FSA Integration Partner

United States Department of Education

Federal Student Aid

FAFSA 7.0 / PIN

ITA Support Report

Final Performance Test Report, Issues List, Conclusions and Recommendations

TO # 102

Deliverable # 102.1.5

Version 1.0

January 31, 2003

Table of Contents

1 Introduction

The U.S. Department of Education's Office Federal Student Aid Programs (FSA) administers and operates the "Free Application for Federal Student Aid" (FAFSA). While available in paper form, FSA also provides this service through a web site. U.S. college students seeking student financial aid use the FAFSA program. During the academic year 2002-2003, over three million students used the web site to apply for federal financial aid. FSA anticipates that the number of users / applicants will increase by 50% during the 2003-2004 academic year, and will continue to rise in future years as the number of paper submissions decreases. This anticipated growth makes it imperative that FSA maximize the capacity and availability of the FAFSA web infrastructure while at the same time minimizing the amount of support FSA's representatives will have to provide for questions by students or difficulties with completing the form. Due to the planned increase in Internet application submissions, an intensive performance testing effort was planned and conducted, as summarized in this document.

PIN is used by Department of Education to associate students with their records. The number of applications that use PIN is growing. FSA anticipates the usage of the PIN application by FAFSA will increase by 50%. There are three major components for PIN: FAFSA PIN, HTML PIN and PIN Web Services. FAFSA PIN is the authentication functionality of FAFSA that allows users to securely access FAFSA business processes. PIN Web Services is authentication used only by the DLSS application. PIN HTML authentication is used by other FSA applications. All three components were performance tested to ensure that they could handle the FAFSA peak load.

2 Executive Summary

2.1 Current Test Results

The purpose of the Performance Test was to identify and resolve bottlenecks in the application, architecture, and infrastructure. The Performance Test also helped determine the optimal configuration for the production environment. In addition, the testing effort helped to determine capacity requirements and to prove that the application would scale and handle the 2003 peak traffic.

All of the goals outlined in the Performance Test Plan were met. There were several key bottlenecks that were identified and resolved, which improved the performance of the application.

2.2 Results Achieved

The performance problems and recommended fixes are categorized by component ranging from Web Server to Mainframe. The specific changes made are included in Section 3 of this document.

- FAFSA Application Code
- IBM HTTP Server - Web Server (IHS)
- IBM WebSphere Application Server (WAS)
- MQ
- Database Access (Oracle and DB2)
- Mainframe
- Infrastructure

In addition to these enhancements, the performance testing resulted in an optimized configuration of the environment.

2.3 Recommendations

The performance testing effort found areas for enhancements to the application, architecture, and infrastructure. Below is the high level list of items that could be evaluated in the future for potential performance improvements to the system:

- Serving HTML images in non-secure way instead of through SSL.
- Reducing session size to 2 ~ 4 KB.
- DB2 lock occurred because test stub was using SSN in sequential order. DRN should be assigned differently.
- Dynamic page caching on WAS is recommended because it minimizes the application server's workload. This will require in depth integration with the development team and should be done during the WebSphere upgrade for FAFSA 8.0.

- Review options to improve CPU utilization of DPL Bridge.
- Upgrade to MQ v5.3 or possibly replace DPL Bridge product with more efficient MQ CICS interface.
- Increasing the planning time for FAFSA application capacity analysis, performance testing, and tuning

The FAFSA application and environment should continue to be tested and tuned. In the future, tests should target an integrated system (possibly in production) with all of the components that make up FAFSA on the Web.

3 Performance Test - High Level Summary

Overall, twenty-three tests were executed from September through the end of December. Each performance test cycle was focused on a specific area of the FAFSA Application. A series of infrastructure tests were run to identify issues with the network, hardware, Web servers, and Application servers. The business process scripts focused on the backend, including WAS, Oracle, DB2, Shadow Direct, MQ, and the Mainframe. The details for each run with conclusions are outlined in Section 4 of this document.

The performance problems and recommended fixes have been outlined below. The recommended performance enhancements are categorized by component ranging from Web Server to Mainframe.

FAFSA Application Code:

- FAFSA Application Code was updated to invalidate sessions. The code update was made to minimize memory usage and disk space in the database.
- The code was updated to resolve invalid time stamps. The invalid time stamps were causing application submissions to fail under heavy load.

IBM HTTP Server - Web Server:

- MaxSpareServers was changed from 100 to 10. The Spare Servers setting was used for handling load spikes. However, leaving a large number of web servers idle, drains system resources. IBM recommended keeping this setting maximized at 10 spare servers.
- MinSpareServers was changed from 10 to 5. The change was made for the same reason as MaxSpareServers.
- MaxRequestPerChild was changed from 10,000 to 0. This setting allows for an upper limit on the number of requests a particular child process can process before it terminates. By setting this to 0, it was ascertained that no individual process could potentially hold all system resources.

IBM WebSphere Application Server:

- Updated HP Kernel Parameters – HP Systems need maintenance on their kernel parameters as a result of these systems' interaction with MQ and WebSphere.
- HP Operating System patches were updated to bring the system up to IBM recommended levels.
- Updated the version of WebSphere to WAS to 3.5.6.
- Increased JVM Memory Size from 512MB to 1GB. This allowed us to better utilize the memory on each application server.
- Decreased number of clones per server from 4 to 2, which allowed us to better utilize the memory on each application server without creating problems with large number of clones.

Database Access:

- A dedicated session database was used for PIN business processes. For a high performance application, it is recommended by IBM, that it have its own session database.
- More space was allocated for the FAFSA session database so that the database will not fill up fast.

MQ:

- CPS region stopped responding to requests when bad data (reversed byte) was sent to the DPL Bridge (bridge between MQ and CICS) and requiring the CICS to be recycled. An IBM patch was applied to fix this issue. The patch allowed the DPL Bridge to handle bad data properly.
- WAS recycled unexpectedly during performance tests. It was determined that AMI, the MQ application interface, caused the recycle. AMI was removed and native Java code was added to interface with MQI. This fixed the WAS recycling issue.
- Connection pooling was added by the EAI team to further enhance the efficiency of MQ Connections. While no performance issues surrounding connections had arisen during initial testing, the creation of pooling provided reuse of existing connections and additional flexibility for the application server connections to the backend. Subsequent testing showed a savings in overall performance by minimizing the number of new connections that needed to be created.
- MQ Queue Managers were not configured to challenge / authenticate new connections therefore the potential existed for unintended traffic from other environments. A security measure was put on MQ queues that controlled who may put messages from specific IP addresses. This helped to minimize the impact of the bad data issue prior to an IBM fix, as it prevented unintended data flow between environments.
- MQ requested CPS Mainframe operator acknowledgement to offload logs. Delays in processing due to this wait, caused errors when operator acknowledgement was not provided in time. In order to resolve this, CPS Mainframe Auto Ops updated the parameter so MQ was permitted to offload automatically rather than requesting a response.
- During data transmission on the bridge between MQ and the mainframe, errors messages CKB5 error messages were observed. These errors would occur under high load and prevent the appropriate expiration of messages. IBM provided a fix to resolve this issue.
- MQ processing was impacted by duplicate messages in the Sync q. This scenario was likely created by an invalid usermod provided by IBM. CSC & EAI ran an IBM provided clean up utility to resolve this issue.
- In production and during backend performance testing a CKB4 Abend – duplicate temp storage entries caused the DPL bridge to Abend. Fix provided by IBM to address issue applied to production.

- In production and during backend performance testing a CKB5 Abend occurred. A unique grouping of persistent and non-persistent messages will caused the DPL Bridge to Abend. Fix provided by IBM to address issue applied.

Mainframe:

All mainframe configuration recommendations are a result of the stub tests and it was not a part of the performance test scope. These recommendations included here are to provide a holistic view of the performance test effort. CSC and Pearson were monitoring the mainframe and DB2 during the performance test.

- A stress test for CICS and DB2 showed that the backend needed to be configured for the FAFSA peak. The following is the recommended DB2 configuration: set Max Task to 75, Thread Limit to 40, and TCB Limit to 50.
- Movement of conversion from CPS mainframe to mid-range servers improved CPS throughput.
- Apply CICS logging usermod APAR UQ62793 which modifies the algorithm used for activity keypointing. This is expected to improve throughput on CPS. This change is scheduled for 2/2/2003.
- The following CICS parameter changes will improve throughput on CPS and is slated for 2/2/2003:
    - AKPFREQ=7500
    - MAXTASK=125
    - DPLBRIDGE tclass=50
    - MAXTCBS=105
    - EDSA=70M
    - THREADLIMIT=50
    - PROTECTNUM=50
    - DPMODE=LOW
- Automation – modify CICSPRD2 USERLOG statement to route message to OPS/MVS for alert notification on MQ. This change will improve notification speed for failures within CPS, and is scheduled for 2/2/2003.
- Automatic start of DPL Bridge via triggering is a change that is expected to minimize user downtime in the event of DPL Bridge failure. This is scheduled for 2/2/2003.

DB2:

- DB2 transactions timed out due to DB2 deadlocks. Pearson effected a code change that prevented table locking.
- Modification of DB2 buffer pools would improve throughput on CPS. The implementation of this change has not been determined. Review is ongoing.

4 Performance Tests – Detailed Description

4.1 Performance Test – Cycle 1

- FAFSA 7.0 Performance Test Cycle 1 was executed on September 26, 2002.
- Hardware & configuration:
  - Web server – HPL14 (4-way, 440 MHz, 8 GB memory. Max web client set to 512)
  - App server – HPN3 (8-way, 360 MHz, 8 GB memory)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: School Code Search
- Goals: The goal of this test cycle was to resolve any environment issues by running one FAFSA business process, School Code Search, in a one web and one application server configuration.

Summary of Results:
In the test, 20 users were added for every 15 seconds ramping up to 500 concurrent virtual users in 6 minutes and 25 seconds. The load was maintained for one hour. As the number of passed transactions increased it was observed that the file FOTW0304.err was over 1GB in size. FOTW0304.err, located in www/fotw/logs is the application log file that has its logging level set to info. In the next test cycles, the logging settings should be increased to the warning or error level to reduce size of this file.

At 500 concurrent virtual users, the CPU Utilization for the web server HPL14 ranged between 30% and 40%. The Memory Utilization averaged at 23%. For the application server, HPN3, the CPU and Memory utilization were 10% and 31% respectively. The response time over the one-hour period remained constant with minor fluctuations. The average response time was 0.34 seconds per transaction. The EAI component, MQ and the DB2 database were also observed at low utilization levels. Message transmitted using MQ was observed at three levels – the queue manager, the bridge manager, and the en-queue message rates. At all monitoring points, there was no queuing or pending messages. For the CICS, there was no backlogging. Processing Utilization levels remained below 1.12% and Disk IO was at 15%. On the CICS environment the shortest response time was observed at 0.012 seconds and the longest response time was observed at 0.5 seconds.

The number of session database records was observed at very high levels. The max number of session database records was set to 5,000. Nevertheless, volumes of 7,971 and 8,209 were observed at the 500 concurrent users level. The number of sessions steadily increased throughout the performance test. It was revealed that JSP pages create sessions by default that were not being invalidated by the application. A JSP directive should be added to the code that will prevent the JSP pages from creating unnecessary sessions.

Conclusions – Cycle 1:

- 500 concurrent users were maintained for 1 hour and the system did not sustain any significant errors.
- ITA recommends that the FOTW0304.err log file be configured to log only error level messages so that the size of the file does not create a disk space issue.
- FAFSA code should be updated to include the directive to prevent JSP pages from creating sessions.

4.2 Performance Test – Cycle 2

Test 1

- FAFSA 7.0 Performance Test Cycle 2 was executed on October 03, 2002.
- Hardware & configuration:
  - Web server – HPL14 (4-way, 440 MHz, 8 GB memory. Max client set to 1024)
  - App server – HPN3 (8-way, 360 MHz, 8 GB memory)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: School Code Search
- Goals: The purpose of this test was to verify that School Code Search was not creating sessions. In Performance Test Cycle 1, the number of session records in the database was at very high levels between 7,000-8,000. A code change was proposed to reduce the number of sessions.

Summary of Results:
In the test, 20 users were added every 15 seconds ramping up to the 1,080 concurrent virtual users. At 1,000 users the number of session records in the database was at 5,419. This value was still high because the directive to prevent JSP pages from creating sessions was not added to the FAFSA 7.0 code. Twenty-five minutes into the test, rapid transaction failures occurred. This was a result of WebSphere shutting down and restarting. The cause for the WebSphere recycle is yet to be determined.

Test 2

- FAFSA 7.0 Performance Test Cycle 2 was executed on October 03, 2002.
- Hardware & configuration:
  - Web server – HPL14 (4-way, 440 MHz, 8 GB memory. Max client set to 1024)
  - App server – HPN3 (8-way, 360 MHz, 8 GB memory)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: Fill Out a FAFSA 7.0
- Goals: The goal is to test the functionality of the Fill Out FAFSA 7.0 business process.

Summary of Results:
After one hour, 700 concurrent virtual users were executing the Fill Out FAFSA business process. Each virtual user would save the form to the TempSave table four times, perform one

school code search, and submit the completed form. There were no environment issues. The test was stopped to proceed to the other goals for Performance Test Cycle 2.

During the test, the response time was constant. From 50 concurrent users submitting the FAFSA 7.0 form to 650 concurrent users, the response time was 0.297 on average. On HPN3 and HPL14 the CPU Utilization levels remained steady at 7% and 31%, respectively. Memory Utilization levels were 31% and 29%, respectively. These levels accurately represent the light load the Fill Out FAFSA was placing on the system resources. There was little activity on the CICS and DB2 backend where response time was 0.016 seconds. There was no queuing in MQ. The active web client connections were well below the 1024 max limit at 153 connections at peak load.

Conclusions – Cycle 2:

- Investigate WebSphere application server recycling problem.
- The JSP directive should be added to School Code Search.
- Fill Out a FAFSA 7.0 business process ran smoothly at 700 concurrent user level.

4.3 Performance Test – Cycle 3

Test 1

- FAFSA 7.0 Performance Test Cycle 3 was executed on October 08, 2002.
- Hardware & configuration:
  - Web server – HPL14 (4-way, 440 MHz, 8 GB memory. Max client set to 1024)
  - App server – HPN3 (8-way, 360 MHz, 8 GB memory)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: School Code Search
- Goals: The purpose of this test was to verify a reduction in the number of database sessions created by School Code Search business process, to investigate the WebSphere recycling issue, and verify a reduction in the number of sessions after adding the JSP directive.

Summary of Results:
In the test, 20 users were added every 15 seconds ramping up to the 500 concurrent virtual users. At this user level, WebSphere recycled. Error messages in the log file showed the command line parameter to disable the "Just-In-Time" compiler was ignored. Additional investigation is required for the WebSphere recycling issue.

The number of session records in the database had successfully been reduced. In previous cycles, at 500 concurrent virtual users level there were 1,732 session records in the database. In this cycle, after the new directive was added to prevent JSP pages from creating a session, there were only 6 session records in the database at the 500 concurrent virtual user level.

Test 2

- FAFSA 7.0 Performance Test Cycle 3 was executed on October 08, 2002.

- Hardware & configuration:
  - Web server – HPL14 (4-way, 440 MHz, 8 GB memory. Max client set to 1024)
  - App server – HPN3 (8-way, 360 MHz, 8 GB memory)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: Fill Out a FAFSA 7.0
- Goals: The purpose of this test was to recreate and resolve the WebSphere production issue of row chaining that occurs when the application tries to write session data in an Oracle database table exceeding the 8K-block size for each row in the table.

Summary of Results:
1,000 concurrent, virtual users were active for this test – 600 executing the Fill Out FAFSA script and 400 executing the JSP script. The JSP script creates a session, 8.036K in size. At the 1,000 users level, 3,300-chained rows were observed. The performance test environment had successfully recreated the production issue.

1,000 concurrent virtual users were active for this retest – 600 executing the Fill Out FAFSA script and 400 executing the JSP script. The purpose of the JSP script was to allow a user to access a JSP page, which created a session 8.036K in size. At the 1,000 users level, 2-chained rows were observed. This was greatly reduced from the previous test with the 8K-block Oracle database where 3,300-chained rows were observed. The performance test environment will now use the 16K block Oracle database for the remaining performance test cycles.

Conclusions – Cycle 3:
- The JSP directive successfully prevented JSP pages from creating sessions. The number of session records was significantly reduced.
- Continue investigation of WebSphere recycling issue. Disabling the "Just-In-Time" compiler did not resolve the issue.
- Successfully recreated and resolved Oracle database the row chaining WebSphere production issue by increasing block size to 16K.

4.4 Performance Test – Cycle 4

Test 1
- FAFSA 7.0 Performance Test Cycle 4 was executed on October 10, 2002.
- Hardware & configuration:
  - Web server – HPL14 (4-way, 440 MHz, 8 GB memory. Max client set to 1024)
  - App server – HPN3 (8-way, 360 MHz, 8 GB memory)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: Fill Out a FAFSA 7.0 - Renewal
- Goals: The goal was to test the functionality and establish a baseline for the Renewal FAFSA 7.0 business processes.

Summary of Results:
500 virtual concurrent users executed the FAFSA Renewal Script. Due to data constraints, each

virtual user would save the form to the TempSave table four times but did not submit the completed form. No environment issues were observed. On HPN3 and HPL14 the CPU utilization levels remained steady at 12% and 28%, respectively. Memory utilization levels were also leveled at 34% and 25%, respectively. There was little activity on the CICS and DB2 backend where DB2 service time was 0.009 seconds. There was no queuing in MQ. The active web client connections were well below the 1024 max limit at 159 connections at peak load.

Test 2

- FAFSA 7.0 Performance Test Cycle 4 was executed on October 10, 2002.
- Hardware & configuration:
  - Web server – HPL14 (4-way, 440 MHz, 8 GB memory. Max client set to 1024)
  - App server – HPN3 (8-way, 360 MHz, 8 GB memory)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: Fill Out a FAFSA 7.0
- Goals: The goal of this test was to recreate the production issue of WebSphere dropping a session table when it is unable to locate a large column in that table.

Summary of Results:
After testing that the WebSphere production issue could be successfully recreated in the performance test environment. It was discovered that the patch IBM provided, debug.jar, did not resolved the issue and WebSphere continued to drop the session table when a large column could not be located. The patch specific to WAS version 3.5.6 did not work properly. ITA worked with IBM to get a new fix (debug.jar) for WAS 3.5.6.

Test 3

- FAFSA 7.0 Performance Test Cycle 4 was executed on October 10, 2002.
- Hardware & configuration:
  - Web server – HPL14 (4-way, 440 MHz, 8 GB memory. Max client set to 1024)
  - App server – HPN3 (8-way, 360 MHz, 8 GB memory)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: Fill Out a FAFSA 7.0
- Goals: The goal of this test was to stress test one clone to determine the maximum number of FAFSA 7.0 applications one clone can submit within a given interval.

Summary of Results:
In a 20-minute span, 1,000 concurrent users were executing Fill Out FAFSA 7.0 using a one web, one application server configuration. At 600 concurrent users, WebSphere recycled resulting in a drop in throughput and hits per second. The test continued and the number of users was ramped up to 1,000. After leveling off at 1,000 users, it was observed that the number of session database connections was at 50, the maximum number allowed. CPU utilization on the web server (HPL14) was at 100%. On the application server, HPN3, CPU utilization was at 41%. Memory utilization levels for both HPN3 and HPL14 were at 30% and 38%, respectively.

Using the Fill Out FAFSA 7.0 business process, the web server was stressed at the 1,000 concurrent user load. Each user, when executing the Fill Out FAFSA application, would save the form to TempSave table four times, perform one school code search, and submit a completed form. At this level, two bottlenecks were discovered -- the number of session database connections and the web server's CPU. With the two bottlenecks ongoing, 28 applications were submitted per minutes on average. The test was stopped to add an additional web server to the architecture and another test was executed with the new configuration.

Test 4

- FAFSA 7.0 Performance Test Cycle 4 was executed on October 10, 2002.
- Hardware & configuration:
  - Web servers – HPL14 & HPL 17 (4-way, 440 MHz, 8 GB memory. Max client set to 1024)
  - App server – HPN3 (8-way, 360 MHz, 8 GB memory)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: Fill Out a FAFSA 7.0
- Goals: The goal of this test was determine the maximum number of applications one clone can process while reducing the utilization levels on the web server by adding a web server to the architecture.

Summary of Results:
In a previous test, 1,000 concurrent virtual users, executing Fill Out FAFSA 7.0, stressed the web server to the point where CPU utilization level was at 100%. After the additional web server was added, the CPU utilization levels dropped considerably.

At the 1,000 concurrent users level, Fill Out FAFSA 7.0 application forms were being submitted via the FAFSA 7.0 application using a two-web, one-application server configuration. At 650 concurrent users, WebSphere recycled resulting in a drop in throughput and the number of hits per second. The test continued and the number of users was ramped up to 1,000. After leveling off at 1,000 users, CPU utilization on the web servers HPL14 and HPL17 were at 10% and 19%, respectively. Memory utilization levels for HPN3, HPL14, HPL17 were at 39%, 30%, and 29%, respectively. The additional web server was able to reduce the high CPU utilization on the web front-end. This test showed the maximum number of application submissions a clone could support was 24 per minute.

Conclusions – Cycle 4

- Successfully tested the Fill Out a FAFSA 7.0 Renewal business process and established a baseline for further testing.
- Tested WebSphere production issue of dropping the session table after applying an IBM-sourced patch. Patch was invalid, further discussions with IBM were conducted.
- Determined the maximum number of application submissions that one application server clone can support is 24 per minute, in this test.

4.5 Performance Test – Cycle 5

Test 1

- FAFSA 7.0 Performance Test Cycle 5 was executed on October 16, 2002.
- Hardware & configuration:
  - Web servers – HPL14 & HPL 17 (4-way, 440 MHz, 8 GB memory. Max client set to 1024)
  - App server – HPN3 (8-way, 360 MHz, 8 GB memory)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: Fill Out a FAFSA 7.0, School Code Search
- Goals: The goal of this test was to investigate and resolve the WebSphere recycling issue.

Summary of Results:
In a pre-test, it was discovered that WebSphere recycling was caused by a memory sharing issue with the MQ AMI interface libraries. The AMI interface was removed and a new set of code was added.

1,000 concurrent users were in the system, 500 users executing the Fill Out FAFSA script and 500 users executing School Code Search. A gradual increase of the number of concurrent user was performed, while utilizing two scripts that had historically led WebSphere to recycle. The first set of 500 concurrent virtual users was held for several minutes before the ramp up was continued. No recycling occurred when ramp up was resumed. After the 1,000 concurrent users level was held. WebSphere did not recycle and each user completed the business process.

During the test, the response time to submit an application was 0.641 seconds at the 1,000 concurrent user level. On HPN3, HPL14, and HPL17 the CPU utilization levels were 20%, 56%, and 42%, respectively. Memory utilization levels were 35%, 30%, and 27%, respectively. There was little activity on the CICS and DB2 backend where response time was 0.017 seconds. MQ, with the new code implemented, functioned as before. Other test cycles will continue to test the WebSphere recycling issue. We could not continue with the other scheduled tests on 10/16/2002 because both the production and development database servers were down.

Test 2

- FAFSA 7.0 Performance Test Cycle 5 was executed on October 18, 2002.
- Hardware & configuration:
  - Web server – HPL14 (4-way, 440 MHz, 8 GB memory. Max client set to 1024)
  - App server – HPN3 (8-way, 360 MHz, 8 GB memory)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: FAA Application, FAA Corrections

- Goals: The goal of this test was to run the FAA Application and FAA Corrections scripts, while testing for potential environment or code issues and establishing a baseline for these scripts.

Summary of Results:
Each FAA Application and FAA Corrections script accesses FAFSA PIN, performs one TempSave and four final checks, and submits the application. In addition, the FAA Corrections functionality uses a Worksheet.

50% of the users were running the FAA Application. The FAA Application ran smoothly without any issue. The other 50% of the users executed FAA Corrections. Only a few users successfully executed FAA Corrections. The remaining users failed at the same point in the script. After investigating this issue, it was discovered that the FAA Correction data had already been submitted and could not be reused. Resubmitting a previously submitted application triggers the following message "Correction is already submitted". The screen was not recorded in the Load Runner script, as a result the users executing FAA Correction failed. The test could not be completed due to the incorrect data provided to ITA and the incorrect scripts. ITA will modify the LoadRunner script and Pearson will provide the data for FAA Correction. This test must be repeated.

Test 3

- FAFSA 7.0 Performance Test Cycle 5 was executed on October 18, 2002.
- Hardware & configuration:
  - Web server – HPL14 (4-way, 440 MHz, 8 GB memory. Max client set to 1024)
  - App server – HPN3 (8-way, 360 MHz, 8 GB memory)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: FAA Application, FAA Corrections
- Goals: The first goal of this test was to determine the number of applications submitted by a clone, and second was to verify the number of connections to the databases.

Summary of Results:
85% of the users were running Fill out a FAFSA and 15% were running FAA Application. This test was executed with one web server and one application server. The Session database connection was set to 100 since the connections reached its limit in the previous test cycle. At 1,000 concurrent users, CPU on the web server was reaching its maximum. CPU utilization on the HPN3 and HPL14 were at 30%, 100%, respectively. Memory utilization levels for HPN3, and HPL14 were at 39% and 41% respectively. The number of Session database connections was still at a maximum of 100. It was observed that on average, 40 applications were submitted per minute. Since the web server CPU was the bottleneck, this test was stopped. The same test scenario will be employed for the next test, with an additional web server.

Test 4

- FAFSA 7.0 Performance Test Cycle 5 was executed on October 18, 2002.

- Hardware & configuration:
  - Web servers – HPL14 & HPL17 (4-way, 440 MHz, 8 GB memory. Max client set to 1024)
  - App server – HPN3 (8-way, 360 MHz, 8 GB memory)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: FAA Application, FAA Corrections
- Goals: The first goal of this test was to determine the number of applications submitted by a clone, and second was to verify the number of connections to the databases. For this test the architecture was modified to include an additional web server.

Summary of Results:
85% of the users were running Fill out a FAFSA and 15% were running FAA Application. This test was executed with two web servers and one application server. There were 1,200 users in the system when a "Null Pointer Error" was observed in the log file. This error was investigated and the EAI team found that the number of channels set in MQI was not sufficient. Also, EAI code needs to be modified to handle the errors properly. The maximum number of channels was set to 100. This was increased to 500. This test needs to be repeated with the above configuration. At 1,200 concurrent users, CPU utilization on the HPN3, HPL14 and HPL17 were at 83%, 27%, and 83%, respectively. Memory utilization levels for HPN3, HPL14, HPL17 were at 38%, 37%, and 39%, respectively. The number of Session database connections was 59, below the maximum value of 100.

Conclusions-Cycle 5

- Successfully accomplished its goal to resolve the WebSphere recycling issue. 1,000 concurrent virtual users executed Fill Out FAFSA and School Code Search processes for 38 minutes and WebSphere did not recycle.
- FAA Application functionality was tested successfully to establish a baseline.
- FAA Corrections test uncovered data and script issues that require rectification.
- Identified a bottleneck on the web server at the 1,000 concurrent user level with one app, one web server configuration. This configuration yielded 40 FAA Application submissions per minute.
- Tested a one-app, two-web server configuration and identified a “Null Pointer Error” in the log file. This error is caused by an insufficient number of MQI channels.

4.6 Performance Test – Cycle 6

Test 1

- FAFSA 7.0 Performance Test Cycle 6 was executed on October 22, 2002.
- Hardware & configuration:
  - Web server – HPL14 (4-way, 440 MHz, 8 GB memory. Max client set to 1024)
  - App server – HPN3 (8-way, 360 MHz, 8 GB memory)
  - Database – DB2 on CICS mainframe environment

  - Messaging – MQ handled traffic between app server and DB2
- Business Process: FAA Application, FAA Corrections
- Goals: The goal of this test was to run the FAA Application and FAA Corrections scripts and check for presence of environment or code issues.

Summary of Results:
Each FAA Application and FAA Corrections script accesses FAFSA PIN, performs one TempSave and four final checks, and submits the application. In addition, the FAA Corrections functionality uses a Worksheet.

User load was distributed evenly between FAA Application and FAA Correction scripts. The test ran smoothly without major issues. Response times in general were averaging 0.454 seconds. Both web and application servers did not appear to be stressed. On HPN3 and HPL14 the CPU utilization levels were 83% and 36%. Memory utilization levels were 34% and 26%, respectively. The execution of this test did not bring forth any script, environment, or application issues.

Tests 2, 3, 4

- FAFSA 7.0 Performance Test Cycle 6 was executed on October 22, 2002.
- Hardware & configuration:
  - Web servers – HPL14 & HPL17 (4-way, 440 MHz, 8 GB memory. Max client set to 1024)
  - App server – HPN3 (8-way, 360 MHz, 8 GB memory)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: Fill Out a FAFSA 7.0, FAA Application
- Goals: The goal of this test was to run the Fill Out FAFSA and FAA Application scripts to determine the number of submissions that can be achieved in a two-web and one application server configuration.

Summary of Results:
70% of the users were running Fill out FAFSA 7.0 and 30% of the users were accessing the FAA Application. This test was run several times. On HPN3, HPL14, and HPL17 the CPU utilization levels were 94%, 39%, and 29%, respectively. Memory utilization levels were 39%, 41%, and 44%, respectively. The session database connections reached its limit.

A zigzag pattern was observed on throughput graph in each test. 50 applications were submitted on average with two web servers and one clone. The rate at which the applications were submitted varied from 14 per minute to 77 per minute. In performance test cycle 6, three different tests were executed to investigate this zigzag pattern. The data from each test is provided. This zigzag pattern needs to be investigated further.

Test 5

- FAFSA 7.0 Performance Test Cycle 6 was executed on October 23, 2002.
- Hardware & configuration:

  - Web servers – HPL14 & HPL17 (4-way, 440 MHz, 8 GB memory. Max client set to 1024)
  - App server – HPN3 (8-way, 360 MHz, 8 GB memory)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: Fill Out a FAFSA 7.0, FAA Application
- Goals: The goal of this test cycle was to troubleshoot the zigzag pattern observed in the throughput graph on the LoadRunner Controller. The test was executed four times each with a different configuration change in order to investigate the zigzag pattern.

Summary of Results:
This test used the Fill Out a FAFSA and FAA Application scripts. The Fill Out a FAFSA script accesses School Code Search once, performs four temporary saves, and submits the application. The FAA Application script accesses FAFSA PIN, performs one temporary save, four final checks, and submits the application.

70% of users were running Fill Out a FAFSA and an additional 30% of the users accessing the FAA Application. For this test, garbage collection was changed to verbose mode. The test ran 800 users smoothly but the duration of garbage collection was 8 seconds. After the garbage collection the number of users was increased to 1020. The zigzag pattern was observed and the session database connections reached its maximum number. On average, 50 applications per minute were submitted. At the 1020 concurrent user level, CPU utilization percentage figures for HPN3 were at 46%, HPL14 at 50%, and HPL17 at 47%. As for Memory utilization: HPN3 stood at 42%, HPL14 at 27%, and HPL17 at 28%.

Test 6

- FAFSA 7.0 Performance Test Cycle 6 was executed on October 23, 2002.
- Hardware & configuration:
  - Web servers – HPL14 & HPL17 (4-way, 440 MHz, 8 GB memory. Max client set to 1024)
  - App server – HPN3 (8-way, 360 MHz, 8 GB memory)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: Fill Out a FAFSA 7.0, FAA Application
- Goals: The goal of this test cycle was to troubleshoot the zigzag pattern observed in the throughput graph on the LoadRunner Controller. The test was executed four times each with a different configuration to investigate and resolve the zigzag pattern.

Summary of Results:
70% of users were running Fill Out a FAFSA and an additional 30% of the users were running the FAA Application. JDBC cumulative efix was applied on the application server and this test was executed with 800 users. The JDBC fix was added to verify whether the duration of garbage collection activity could be reduced to levels observed during last year's test. The test ran smoothly but the duration of garbage collection remained at 8 seconds.

After the garbage collection the number of users were increased to 1020. For the application server, HPN3, CPU capacity was maximized and the saturation point was reached under this configuration. As a result, the response time had increased. The session database connection was also maximized. The zigzag pattern witnessed in test 1 reappeared. Application submission stood at an average rate of 33 per minute. Prior to the CPU's saturation point the following figures were obtained for utilization percentage: For HPN3 it was 38%, HPL14 was at 57%, and HPL17 was 52%. Memory utilization for HPN3 was 37%, HPL14 27%, and HPL17 was 28%. CPU on the application server was a bottleneck with 1,000 concurrent users in this configuration. The zigzag pattern still requires further investigation.

Test 7

- FAFSA 7.0 Performance Test Cycle 6 was executed on October 23, 2002.
- Hardware & configuration:
  - Web servers – HPL14 & HPL17 (4-way, 440 MHz, 8 GB memory. Max client set to 1024)
  - App server – HPN3 (8-way, 360 MHz, 8 GB memory)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: Fill Out a FAFSA 7.0, FAA Application
- Goals: The goal of this test cycle was to troubleshoot the zigzag pattern observed in the throughput graph on the LoadRunner Controller.

Summary of Results:
The JVM Heap size was decreased from 1024MB to 512MB and this test was executed with 800 users. No environment issues were observed when a garbage collection took 3.7 seconds. After the garbage collection the number of users were increased to 900. The application server was running about 30% on CPU and no environment issues were observed. Garbage collection was taking between 3.7 to 5 seconds to complete. The zigzag pattern was not observed in this test. Reducing the JVM Heap size reduced the zigzag pattern in the throughput. 50 applications per minute were submitted on average. Final CPU utilization percentage figures for HPN3 were at 74%, HPL14 at 32%, and HPL17 at 41%. As for Memory utilization: HPN3 stood at 36%, HPL14 at 27%, and HPL17 at 28%.

Test 8

- FAFSA 7.0 Performance Test Cycle 6 was executed on October 23, 2002.
- Hardware & configuration:
  - Web servers – HPL14 & HPL17 (4-way, 440 MHz, 8 GB memory. Max client set to 1024)
  - App servers – HPN3 (8-way, 360 MHz, 8 GB memory), HPN8 (8-way, 750 MHz, 8 GB memory)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: Fill Out a FAFSA 7.0, FAA Application
- Goals: The goal of this test cycle was to determine the number of applications submitted by one clone per application server.

Summary of Results:
An additional application server, HPN8 was added to the performance test environment to determine the number of applications that can be submitted by two clones. The test could not be completed because one of the application servers ran out of disk space. The test needs to be repeated.

Conclusions-Cycle 6

- Tested FAA Application and FAA Corrections functionality successfully.
- For tests 2, 3, and 4 a cyclical pattern in the throughput graph coupled with high response time were observed and investigated.
- The zigzag pattern was observed and elevated levels of the session database connections occur after a garbage collection.
- The JDBC fix was added to verify whether the duration of garbage collection activity could be reduced. Garbage collection time remained at 8 seconds.
- JVM heap size was reduced from 1024 MB to 512 MB and an application server was added, but zigzag pattern in throughput graph persisted, hence further testing is necessary.

4.7 Performance Test – Cycle 7

Test 1

- FAFSA 7.0 Performance Test Cycle 7 was executed on October 24, 2002.
- Hardware & configuration:
  - Web servers – HPL14 & HPL17 (4-way, 440 MHz, 8 GB memory. Max client set to 1024)
  - App servers – HPN3 (8-way, 360 MHz, 8 GB memory), HPN8 (8-way, 750 MHz, 8 GB memory)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: PIN Authentication
- Goals: The first goal of this test was to determine number of transactions that could be achieved in a minute for the PIN business process while using two web servers and two application servers, while the second goal was to determine if there were any environment and code issues with the FAFSA PIN Authentication script.

Summary of Results:
This test ran with 500 users and the load was increased to 1500 users. This test ran for 50 minutes and a total of 30,784 authentication transactions completed (i.e. 30,784 users completed authentications in 50 minutes). CPU Utilization on web servers fluctuated between 70% and 100%. At 1,500 users, CPU utilization levels were at 11% on HPN3, 7% on HPN8, 92% on HPL14, and 100% on HPL17. Memory Utilization levels were at 37% on HPN3, 29% on HPN8, and 48% on both HPL14 and HPL17. CICS and MQ did not receive any traffic in this test. No code and environment issues were observed.

Test 2

- FAFSA 7.0 Performance Test Cycle 7 was executed on October 24, 2002.
- Hardware & configuration:
  - Web servers – HPL14 & HPL17 (4-way, 440 MHz, 8 GB memory. Max client set to 1024)
  - App server – HPN3 (8-way, 360 MHz, 8 GB memory)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: PIN Authentication, PIN HTML Authentication
- Goals: The first goal of this test was to determine number of transactions that can be achieved in a minute for this business processes while using two web servers and one application server. The second goal of this test cycle focused on determining whether or not environment and code issues existed with the FAFSA PIN Authentication and PIN HTML Authentication scripts.

Summary of Results:
50% of the users executed PIN HTML Authentication and the other 50% executed PIN Registration. Some PIN Registration users failed. Pearson discovered that the failure was due to a column missing in the CPS table. This column was added and the test was repeated. Both business processes ran successfully. PIN was using WAS admin database as a session database which is not a good practice for a high volume application. A new dedicated PIN session database needs to be in place for PIN. ITA has requested a dedicated session database for PIN from CSC.

This test ran with 500 users and the load was increased to 1500 users. CPU on the web servers fluctuated between 75% and 90%. At 1,500 users, CPU utilization levels were at 26% on HPN3, 87% on HPL14, and 87% on HPL17. Memory Utilization levels were at 36% on HPN3, 39% on HPL14, and 41% on HPL17. No code and environment issues were observed. More than 10,000 transactions were completed per hour on the PIN Registration and more than 20,000 transactions per hour on the HTML PIN Authentication.

Test 3

- FAFSA 7.0 Performance Test Cycle 7 was executed on October 24, 2002.
- Hardware & configuration:
  - Web servers – HPL14 & HPL17 (4-way, 440 MHz, 8 GB memory. Max client set to 1024)
  - App servers – HPN3 (8-way, 360 MHz, 8 GB memory), HPN8 (8-way, 750 MHz, 8 GB memory)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: Fill Out a FAFSA 7.0, FAA Application
- Goals: The goal of this test was to determine the number of applications submitted by two web servers and two application servers.

Summary of Results:
Each application server had one clone, a heap size of 512MB, and 100 maximum connections to the session database.

70% of the users executed Fill out FAFSA and 30% executed FAA Application. On average, 95 applications per minute were being saved temporarily and 80 applications were being submitted per minute. At 1,500 users, CPU utilization levels were at 40% on HPN3, 14% on HPN8, 96% on HPL14, and 96% on HPL17. Memory Utilization levels was at 37% on HPN3, 30% on HPN8, and 33% on both HPL14 and HPL17. Garbage collection was taking 3.7 to 7 seconds. The garbage collection time appears to be somewhat long and this needs to be investigated further.

Conclusions-Cycle 7

- Successfully tested PIN Authentication functionality.
- Successfully test PIN Authentication and PIN HTML Authentication functionality.
- Tested Fill Out a FAFSA 7.0 and FAA Application scripts. Duration of garbage collection needs further reduction.

4.8 Performance Test – Cycle 8

Test 1

- FAFSA 7.0 Performance Test Cycle 8 was executed on October 30, 2002.
- Hardware & configuration:
  - Web server – HPL14 (4-way, 440 MHz, 8 GB memory. Max client set to 1024)
  - App server – HPN3 (8-way, 360 MHz, 8 GB memory)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: Fill Out a FAFSA 7.0
- Goals: The goal this test was to recreate the production issue of WebSphere dropping a session table when it is unable to locate a large column in that table.

Summary of Results:
After successful recreation of the WebSphere production issue in the performance test environment, the efix supplied by IBM was applied. The test was repeated with the fix in place and the session table did not drop. The efix provided by IBM prevents the session table from dropping even if WebSphere gets a bad response from SQL Query.

In production when the session table was dropped - the application servers were restarted to bring PIN into the WebSphere environment. The clones were initializing when the application server was restarted. When the clones were initializing WebSphere queried the session database (medium and large column in the session table) and received a bad response. Because of that bad response from the query, WebSphere dropped the session database. There is not enough information to determine why the bad query occurred in production. There was a

message in a log file indicating that WebSphere is dropping the session database. The efix provided by IBM does not permit WebSphere to drop the session table regardless of the response it gets from the SQL query. This issue is closed from the performance test perspective however, ITA will monitor the performance test and in production (during peak especially, if the application servers are stopped and started), for potential occurrences of this issue.

Test 2

- FAFSA 7.0 Performance Test Cycle 8 was executed on October 30, 2002.
- Hardware & configuration: The configuration for the test was a PIN test harness that was linked to the performance test architecture - one web server and one application server.
  - Web server – HPL14 (4-way, 440 MHz, 8 GB memory. Max client set to 1024)
    - Web server – PIN test harness – Su35e6 (4-way, 400 MHz, 4GB memory)
  - App server – HPN3 (8-way, 360 MHz, 8 GB memory)
    - App server – PIN test harness – Su35e11 (4-way, 400 MHz, 4GB memory)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: PIN Web Services Authentication
- Goals: The first goal of this test was to run PIN Web Services to ensure there were no code, environment, or script issues. The second goal was to capture the response time on Web Services. A PIN Web Services test harness was created in order to execute the test.

Summary of Results:
500 users ran smoothly with PIN Web Services. This business process was creating sessions in the session database. The sessions were being created from (rpcrouter) that could not be invalidated because the Web Service does not have access to the request object. There are a couple of options / workarounds to resolve this issue:

- Modify the rpcrouter servlet.
- Investigate if the Web Services can invalidate the sessions.
- Place Web Services in its own web application with 2 minutes session timeout.

It is recommended that a separate web application be created for PIN Web Services with 2 minutes session timeout and repeat this test. This performance test can be combined with other PIN business processes (HTML PIN Authentication and PIN registration). At 500 users, CPU utilization levels were at 7% on HPN3 and 3% on HPL14. Memory Utilization levels were 32% on HPN3 and 22% on HPL14.

Test 3

- FAFSA 7.0 Performance Test Cycle 8 was executed on October 30, 2002.
- Hardware & configuration: The configuration for the test was a PIN test harness that was linked to the performance test architecture- one web server and one application server.

- Web servers – HPL14 & HPL 17 (4-way, 440 MHz, 8 GB memory. Max client set to 1024)
  - Web server – PIN test harness – Su35e6 (4-way, 400 MHz, 4GB memory)
- App server – HPN3 (8-way, 360 MHz, 8 GB memory)
  - App server – PIN test harness – Su35e11 (4-way, 400 MHz, 4GB memory)
- Database – DB2 on CICS mainframe environment
- Messaging – MQ handled traffic between app server and DB2

- Business Process: PIN Web Services Authentication, PIN Registration, PIN HTML Authentication
- Goals: The goal of this test was to determine number of transactions that could be achieved in one minute for these business processes while using two web servers and one application server.

Summary of Results:
The number of users was distributed equally among the three business processes. At 1,000 users, CPU utilization levels were at 10% on HPN3, 32% on HPL14, and 39% on HPL17. Memory utilization levels were at 33% on HPN3, 26% on HPL14, and 30% on HPL17. Since SU35E11 did not respond after 20 minutes in the test - this issue needs to be investigated further and the test needs to be repeated.

Test 4

- FAFSA 7.0 Performance Test Cycle 8 was executed on October 30, 2002.
- Hardware & configuration:
  - Web server – HPL14 (4-way, 440 MHz, 8 GB memory. Max client set to 1024)
  - App server – HPN3 (8-way, 360 MHz, 8 GB memory)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: FAFSA Corrections 7.0
- Goals: The goal of this test was to run FAFSA Corrections 7.0 to ensure there were no code, environment, or script issues.

Summary of Results:
This test was executed several times. The Mainframe tables were loaded with 750,000 records to prepare for the Mainframe load testing. After the data were loaded, the plans were not rebound as the old access path was used that had bad indexes (the indexes from when the tables were only 20,000 records). During this test, the response time was poor and FAFSA Corrections 7.0 records were failing because of these bad access paths. The Mainframe team rebound the plans and during the re-bind errors were observed. Once the rebind was completed, the number of errors started to decrease. This test was stopped and repeated.

In the next test, there were deadlocks in DB2. EAI exception was also observed in the log file indicating that the response timed out. The FAFSA Corrections 7.0 records were failing. An error message was seen in the log file "EAI exception occurred" - it was an EAI timeout message. The test was executed a third time and the same pattern was observed – FAFSA

Corrections 7.0 records were failing, there were deadlocks in DB2 and an EAI error message in the log file. This issue needs to be investigated further.

At 500 users, CPU utilization levels were at 30% on HPN3 and 46% on HPL14. Memory utilization levels were at 34% on HPN3 and 26% on HPL14.

Test 5

- FAFSA 7.0 Performance Test Cycle 8 was executed on October 31, 2002.
- Hardware & configuration:
  - Web servers – HPL14 & HPL17 (4-way, 440 MHz, 8 GB memory. Max client set to 1024)
  - App server – HPN3 (8-way, 360 MHz, 8 GB memory)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: FAFSA Corrections 7.0
- Goals: The goal of this test cycle was to troubleshoot the DB2 deadlocks observed in a test Cycle 8. The test was executed twice each time with a different configuration in order to investigate this DB2 issue.

Summary of Results:
750,000 rows were backed out from DB2. The rows were added on 10/30 for the mainframe load test. During the test, EAI Code detected no response from FAFSA. There was no error message in the log file. DB2 deadlocks were observed. It is recommended that the mainframe should be loaded to run this test. At the 600 concurrent users level, CPU utilization percentage figures for HPN3 were 28%, HPL14 27%, and HPL17 26%. As for Memory utilization: HPN3 were 34%, HPL14 24%, and HPL17 26%.

Test 6

- FAFSA 7.0 Performance Test Cycle 8 was executed on October 31, 2002.
- Hardware & configuration:
  - Web servers– HPL14 & HPL17 (4-way, 440 MHz, 8 GB memory. Max client set to 1024)
  - App servers – HPN3 (8-way, 360 MHz, 8 GB memory), HPN8 (8-way, 750 MHz, 8 GB memory)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: FAFSA Corrections 7.0
- Goals: The goal of this test cycle was to troubleshoot the DB2 deadlocks observed in a previous test cycle.

Summary of Results:
The configuration tested was two web servers and application server HPN3 executing FAFSA Corrections 7.0. The application server was changed from HPN3 to HPN8. The same results were observed, in that, DB2 deadlocks, were observed in this test.

Test 7

- FAFSA 7.0 Performance Test Cycle 8 was executed on October 31, 2002.
- Hardware & configuration:
  - Web servers– HPL14 & HPL17 (4-way, 440 MHz, 8 GB memory. Max client set to 1024)
  - App server – HPN3 (8-way, 360 MHz, 8 GB memory)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: Fill Out a FAFSA 7.0
- Goals: The goal of this test cycle was to execute Fill out a FAFSA 7.0 and focusing on the issue of session database connections increasing after a garbage collection occurs.

Summary of Results:
This test was executed several times with different configurations. The persistence session was turned off to see if the garbage collection time would change. It was observed that the garbage collections took 4.9 seconds (garbage collection time did not change).

Test 8

- FAFSA 7.0 Performance Test Cycle 8 was executed on October 31, 2002.
- Hardware & configuration:
  - Web servers– HPL14 & HPL17 (4-way, 440 MHz, 8 GB memory. Max client set to 1024)
  - App server – HPN8 (8-way, 750 MHz, 8 GB memory)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: Fill Out a FAFSA 7.0
- Goals: The goal of this test cycle was to execute Fill out a FAFSA 7.0 and focusing on the issue of session database connections increasing after a garbage collection occurs.

Summary of Results:
This test was executed several times with different configurations. In this test, the application server HPN3 was replaced with HPN8. Session database connections increased significantly, and garbage collections took 4.9 seconds (garbage collection time did not change).

Test 9

- FAFSA 7.0 Performance Test Cycle 8 was executed on October 31, 2002.
- Hardware & configuration:
  - Web servers– HPL14 & HPL17 (4-way, 440 MHz, 8 GB memory. Max client set to 1024)
  - App server – HPN8 (8-way, 750 MHz, 8 GB memory)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: Fill Out a FAFSA 7.0

- Goals: The goal of this test cycle was to execute Fill out a FAFSA 7.0 and focusing on the issue of session database connections increasing after a garbage collection occurs.

Summary of Results:
This test was executed several times with different configurations. In this test, an efix was added to the application server, HPN8. This efix removes prepared statement caching. The same results were observed - session database connections jumped significantly. For this test, all the logs and necessary information were collected. ITA will work with IBM support on this issue.

Test 10

- FAFSA 7.0 Performance Test Cycle 8 was executed on October 31, 2002.
- Hardware & configuration:
  - Web servers– HPL14 & HPL17 (4-way, 440 MHz, 8 GB memory. Max client set to 1024)
  - App server – HPN8 (8-way, 750 MHz, 8 GB memory)
  - Database – DB2 on CICS mainframe environment
  - Messaging – Shadow Direct handled traffic between app server and DB2
- Business Process: Fill Out a FAFSA 7.0
- Goals: The goal of this test cycle was to execute Fill out a FAFSA 7.0 and focusing on the issue of session database connections increasing after a garbage collection occurs.

Summary of Results:
This test was executed several times with different configurations. In this test, the configuration was switched from MQ to Shadow Direct. The same results were observed – the session database connections jumped significantly.

Conclusions-Cycle 8

- The production issue was on FAFSA 6.0 recreated successfully. IBM-supplied efix precluded session table from being dropped, even when WebSphere receives a bad response from an SQL Query.
- PIN Web Services Authentication test is in progress. PIN Web Services Authentication process was creating sessions that were not being invalidated. Several recommendations were presented to resolve issues encountered during the test.
- PIN Registration, PIN HTML Authentication, and PIN Web Service Authentication business processes were tested. The test is in progress. The PIN test harness that resides on SU35E6 and SU35E11 ceased to respond after 20 minutes of testing hence the test was stopped. Further investigation is needed.
- FAFSA Corrections 7.0 business process was tested. This is test is in progress. DB2 deadlocks were observed during the test and after the garbage collection, the number of session database connections increased significantly.

- The Fill Out a FAFSA 7.0 business process was tested for a number of performance tests utilizing a variety of hardware configurations, all of which focused on troubleshooting the issue of increasing number of session database connections. This condition occurred after a garbage collection. An efix was applied to HPN8 that removes prepared statement caching, but it did not resolve the issue. MQ was replaced with Shadow Direct, but the same result was observed. ITA has collected information that will be shared with IBM support.

4.9 Performance Test – Cycle 9

Test 1

- FAFSA 7.0 Performance Test Cycle 9 was executed on November 5, 2002.
- Hardware & configuration:
  - Web servers– HPL14 (4-way, 440 MHz, 8 GB memory. Max client set to 1024)
  - App server – HPN3 (8-way, 360 MHz, 8 GB memory)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: Fill Out a FAFSA 6.0
- Goals: The goal of this test cycle was to run the Fill Out a FAFSA 6.0 script to investigate potential issues with LoadRunner script, application code, and the environment.

Summary of Results:
The Fill Out a FAFSA 6.0 script accesses School Code Search once, performs four temporary saves, and one application submission step.

The test ran 500 users smoothly, without issues, and the garbage collection lasted 4 seconds. Over the course of the 35-minute test, 688 applications were submitted with an average of 19.65 applications per minute. The number of session database connections remained steady at 25 and did not increase. At 500 concurrent virtual users, HPN3 and HPL14 CPU utilization levels were 19% and 71%, respectively; Memory utilization for HPN3 and HPL14 was 37% and 28%, respectively.

Test 2

- FAFSA 7.0 Performance Test Cycle 9 was executed on November 5, 2002.
- Hardware & configuration:
  - Web servers– HPL14 (4-way, 440 MHz, 8 GB memory. Max client set to 1024)
  - App server – HPN3 (8-way, 360 MHz, 8 GB memory)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: FAFSA Corrections 7.0
- Goals: The first goal of this test is to run FAFSA Corrections 7.0 script to investigate potential issues with the application code, environment, or the script. The second goal

was to focus on DB2 deadlocks and session database connections. Load was placed on the Mainframe to measure the CPS performance.

Summary of Results:
The test was executed twice. NCS generated load on the mainframe for this test. An EAI exception was observed in the log file indicating that no reply was received within specified timeframe. The LoadRunner virtual users were failing. After some investigation it was found that the batch jobs that were running on the mainframe had problems with inserting records and committing them. DB2 deadlocks were observed, as well.

The batch jobs were fixed and the test was executed a second time. The same errors were observed - LoadRunner users failed, DB2 locks were observed, and a timeout message was seen in the log file. At this time it was discovered that the batch jobs were using data from last year, which was causing this problem. The test was stopped. Once the batch jobs are fixed, this test will be repeated.

Test 3

- FAFSA 7.0 Performance Test Cycle 9 was executed on November 5, 2002.
- Hardware & configuration:
  - Web servers– HPL14 & HPL17 (4-way, 440 MHz, 8 GB memory. Max client set to 1024)
  - App servers – HPN3 (8-way, 360 MHz, 8 GB memory), HPN 8 (8-way, 750 MHz, 8 GB memory)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: FAFSA TempSave-Restore, Fill Out a FAFSA 6.0, Fill Out a FAFSA 7.0, and FAFSA Corrections 7.0
- Goals: The goal of this test was to determine the number of temporary saves and the number of applications submitted by two web servers and two application servers.

Summary of Results:
Each application server clone has a heap size of 512 MB, and a maximum session DB connection setting of 100.

No load was generated on the mainframe, as there were problems with the batch process. The test started with 400 users and increased to 1025 users. During the test, the garbage collection took about 3.9 seconds and the session database connections did not increase rapidly but held steady at 49 connections. CPU and memory stats were as follow: HPN3 -- 36% CPU and 38% memory; HPN8 -- 12% CPU and 32% memory; HPL14 - 80% CPU and 32% memory; HPL17 80% CPU and 27% memory.

CPU on the web servers, the HP L-class machines, was fluctuating between 70% and 95%. Web Servers' CPU became the bottleneck for this test. 23 applications were submitted per minute and 73 temporary saves were submitted per minute.

Test 4

- FAFSA 7.0 Performance Test Cycle 9 was executed on November 6, 2002.
- Hardware & configuration:
  - Web servers– HPL14 & HPL17 (4-way, 440 MHz, 8 GB memory. Max client set to 1024)
  - App servers – HPN3 (8-way, 360 MHz, 8 GB memory), HPN 8 (8-way, 750 MHz, 8 GB memory)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: FAFSA TempSave-Restore, Fill Out a FAFSA 6.0, Fill Out a FAFSA 7.0, and FAFSA Corrections 7.0
- Goals: The goal of this test was to determine the number Temporary Saves and number of applications submitted by two web servers and two application servers.

Summary of Results:
There is one clone on each application server. This test was executed several times changing one parameter at a time. This test was the baseline test with no configuration changes.

The test ramped up to 1025 concurrent virtual users. CPU utilization levels were 39% for HPN3, 21% for HPN8, 76% for HPL14, and 71% for HPL17. Memory utilization levels were 39% for HPN3, 34% for HPN8, 45% for HPL14, and 43% for HPL17. At this user level, it was observed that garbage collection was taking place frequently (every 11 seconds). Therefore the test was stopped and the heap size was increased from 512MB to 1GB. This test was repeated.

Test 5

- FAFSA 7.0 Performance Test Cycle 9 was executed on November 6, 2002.
- Hardware & configuration:
  - Web servers– HPL14 & HPL17 (4-way, 440 MHz, 8 GB memory. Max client set to 1024)
  - App servers – HPN3 (8-way, 360 MHz, 8 GB memory), HPN 8 (8-way, 750 MHz, 8 GB memory)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: FAFSA TempSave-Restore, Fill Out a FAFSA 6.0, Fill Out a FAFSA 7.0, and FAFSA Corrections 7.0
- Goals: The goal of this test was to determine the number of Temporary Saves and number of applications submitted by two web servers and two application servers.

Summary of Results:
This test was executed several times changing one parameter at a time. The heap size configuration was changed from 512Mb to 1GB. At 1000 virtual concurrent users, the CPU on the web servers were bottlenecks. CPU utilization levels were at 30% for HPN3, 15% for HPN8, 93% for HPL14, and 100% for HPL17. Memory utilization levels were at 43% for HPN3, 37% for HPN8, 44% for HPL14, and 44% for HPL17. 21 applications were submitted per minute and 74 temporary saves per minute.

Test 6

- FAFSA 7.0 Performance Test Cycle 9 was executed on November 6, 2002.
- Hardware & configuration:
  - Web servers– HPL14 & HPL17 (4-way, 440 MHz, 8 GB memory. Max client set to 1024)
  - App servers – HPN3 (8-way, 360 MHz, 8 GB memory), HPN 8 (8-way, 750 MHz, 8 GB memory)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: FAFSA TempSave-Restore, Fill Out a FAFSA 6.0, Fill Out a FAFSA 7.0, and FAFSA Corrections 7.0
- Goals: The goal of this test was to determine the number of Temporary Saves and number of applications submitted by two web servers and two application servers.

Summary of Results:
Since the web servers CPUs were the bottleneck, the test was executed with the Secure Sockets Layer protocol (SSL) turned off, and the test was run once again with 1000 users. CPUs and memory stats were as follows: HPN3 - 44% CPU and 46% memory; HPN8 - 12% CPU and 40% memory; HPL14 - 70% CPU and 39% memory; HPL17 - 88% CPU and 38% memory.

Application submission reached 23 applications per minute and TempSave per minute were 68. The number of users was increased to 1200. At this threshold, the number of applications submitted was 24 per minute and temporary saves per minute reached 75. The CPU utilization rates on web servers were 88% and 90%. Thus, turning SSL did not significantly reduce CPU utilization for the web server. It was also observed that the number of daemons was peaking on the web servers. Thus, the test was stopped and the threshold for daemons was reset from 1024 to 2048.

Conclusions-Cycle 9

- Fill Out a FAFSA 6.0 functionality was tested successfully.
- FAFSA Corrections 7.0 was not completed due to problems with batch jobs running on the mainframe that led to failure of virtual users generated by LoadRunner.
- Running FAFSA TempSave-Restore, Fill Out a FAFSA 6.0, Fill Out a FAFSA 7.0, and FAFSA Corrections 7.0 scripts for Test 3 led to CPU fluctuations on the web servers. However, garbage collection time was approximately 3.9 seconds and the number of session database connections did not increase rapidly.
- For tests 4, 5, 6 the same scripts were run (FAFSA TempSave-Restore, Fill Out a FAFSA 6.0, Fill Out a FAFSA 7.0, and FAFSA Corrections 7.0) and each test focused on single parameter change in order to isolate the problem of fluctuating CPU values. The JVM size was increased from 512 MB to 1GB, SSL was turned off, and the number of daemons increased from 1024 to 2048. It was also noticed that FAFSA is serving all the images as SSL.

4.10 Performance Test – Cycle 10

Test 1

- FAFSA 7.0 Performance Test Cycle 10 was executed on November 7, 2002.
- Hardware & configuration:
  - Web server – HPL14 (4-way, 440 MHz, 8 GB memory. Max client set to 1024)
  - App server – HPN3 (8-way, 360 MHz, 8 GB memory)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: Student Access
- Goals: The goal of this test was investigate potential code, environment, or script issues for the Student Access business process. 500 users load was placed on the application.

Summary of Results:
The CPU utilization on the application server was high at 90%. 500 users were applied to the system during the test. The session size created by Student Access was about 18 KB and the session database that was used had 16KB block size. Row chaining occurred as the result. A handful of options were investigated for this issue (e.g., reducing session size, increasing database space).

Test 2

- FAFSA 7.0 Performance Test Cycle 10 was executed on November 7, 2002.
- Hardware & configuration:
  - Web server – HPL14 (4-way, 440 MHz, 8 GB memory. Max client set to 1024)
  - App server – HPN3 (8-way, 360 MHz, 8 GB memory)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: FAFSA Corrections 7.0
- Goals: The goal of this test was to further investigate the FAFSA Corrections 7.0 business process errors that appeared during the previous cycles. It was noted that errors occurred when batch jobs were run on CPS mainframe. Thus, batch jobs were scheduled and set to run during this cycle.

Summary of Results:
A 500-user load was generated on FAFSA Corrections 7.0. During the test, LoadRunner virtual users were failing and the timeout message appeared in the log file. The application timeout interval was shorter than the DB2 lock wait timeout interval of 120 seconds. There was lock contention between the "correct FAFSA" script and the batch "compute" process. The following message appeared in the CICS log " TRAN W412 CODE 0000 FROM MSG F50-UNEXPECTED ERROR RETURNED FROM A40-DE TRAN W412 CODE .... FROM MSG W12:". The batch process will be investigated and once it is fixed this test will be run again.

Test 3

- FAFSA 7.0 Performance Test Cycle 10 was executed on November 7, 2002.
- Hardware & configuration:
  - Web server – HPN8 (8-way, 750 MHz, 8 GB memory. Max client set to 1024)
  - App server – HPN3 (8-way, 360 MHz, 8 GB memory)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: FAFSA TempSave-Restore, Fill Out a FAFSA 6.0, Fill Out a FAFSA 7.0, and FAFSA Corrections 7.0
- Goals: The goal of this test was to determine the number of application submissions and temporary saves for a given application server.

Summary of Results:
In the previous cycles, a HP L-class machine was used as the web server. L-class servers are less powerful than the N-class machines and thus caused a bottleneck for the performance test. By using HPN8 as the web server a more accurate measurement can be taken for capacity planning analysis.

Performance gains were observed by using HPN8 as the web server. At 1000 concurrent user level, the web server CPU was at 50% utilization. Response times were generally low before reaching 1000 concurrent users. At 1000 concurrent users level, response time on average was at 30 seconds. Delays on the application server side were observed and a decision was made to increase the thread count up to 75 from 25.

Test 4

- FAFSA 7.0 Performance Test Cycle 10 was executed on November 7, 2002.
- Hardware & configuration:
  - Web server – HPN8 (8-way, 750 MHz, 8 GB memory. Max client set to 1024)
  - App server – HPN3 (8-way, 360 MHz, 8 GB memory)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: FAFSA TempSave-Restore, Fill Out a FAFSA 6.0, Fill Out a FAFSA 7.0, and FAFSA Corrections 7.0
- Goals: The goal of this test was to determine the number of application submissions and Temporary Saves for a given application server.

Summary of Results:
In the previous cycles, a HP L-class machine was used as the web server. L-class servers are less powerful than the N-class machines and caused a bottleneck for the performance test. By using HPN8 as the web server, the server bottleneck issue should be resolved. Also, a second application server clone was added for this test. The intention was to ease the load on one application server and to monitor the behavior of two clones.

Response time was generally low and averaging about 5 seconds. The CPU on HPN3 was at

50% while memory was at 53%. For HPN8, CPU was at 60% and memory at 53%. Due to testing time frame issue, the test was stopped before a conclusive result could be gained. The test will be executed again in future cycles.

Test 5

- FAFSA 7.0 Performance Test Cycle 10 was executed on November 8, 2002.
- Hardware & configuration:
  - App server – HPN3 (8-way, 360 MHz, 8 GB memory)
  - Web server – HPN8 (8-way, 750 MHz, 8 GB memory. Max client set to 1024)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: FAFSA TempSave-Restore, Fill Out a FAFSA 6.0, Fill Out a FAFSA 7.0, and FAFSA Corrections 7.0
- Goals: The goal of this test was to determine the number of applications and Temporary Saves submitted by one web server and two-application server clones.

Summary of Results:
Each clone had 1,024 MB heap size, 100 maximum session DB connections, and 75 maximum thread count. The Web server Max Clients setting was set to 1024.

Hits per second was about half of what was achieved in the last test of cycle 10. Transaction response time also grew to 30 seconds. It was also observed that Web the Server Max Clients reached its limit. It can be concluded that the Web Server Max Clients setting was responsible for the low throughput and should be increased back to 2048.

Test 6

- FAFSA 7.0 Performance Test Cycle 10 was executed on November 8, 2002.
- Hardware & configuration:
  - Web server – HPN8 (8-way, 750 MHz, 8 GB memory. Max client set to 2048)
  - App server – HPN3 (8-way, 360 MHz, 8 GB memory)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: FAFSA TempSave-Restore, Fill Out a FAFSA 6.0, Fill Out a FAFSA 7.0, and FAFSA Corrections 7.0
- Goals: The goal of this test was to determine the number of applications and Temporary Saves submitted by one web server and two-application server clones.

Summary of Results:
Each application server clone had a 512 MB heap size, 100 maximum session DB connections, and 75 maximum thread count.

Throughput was back to the 40 Mb per second range while the figure for hits per second was 2,110. Transaction response time was generally below 10 seconds. After 28 minutes had elapsed from the beginning of the test, a significant number of exceptions occurred. It was observed that a network communication error occurred causing time outs on all transactions.

Error message returned: A communications error for TCP / IP occurred.
Explanation: An unexpected error occurred in communications.
Action: The return code from the TCP / IP (select) [TIMEOUT] 360 seconds call was 246.

CPU utilization levels for HPN3 and HPN8 were 73% and 61%, respectively. Memory utilization levels were 53% and 59%, respectively.

Test 7

- FAFSA 7.0 Performance Test Cycle 10 was executed on November 8, 2002.
- Hardware & configuration:
  - Web server – HPN8 (8-way, 750 MHz, 8 GB memory. Max client set to 2048)
  - App server – HPN3 (8-way, 360 MHz, 8 GB memory)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: FAFSA TempSave-Restore, Fill Out a FAFSA 6.0, Fill Out a FAFSA 7.0, and FAFSA Corrections 7.0
- Goals: The goal of this test was to determine the number of applications and temporary saves submitted by two web servers and two-application server clones.

Summary of Results:
Each clone had 512 MB heap size, 100 session DB connections, and 75-thread count. The Web server Max Clients setting was set to 1024.

This test was run for one hour. Throughput was at 40 Mb / second with 2,116 hits / second. Transaction response time was generally below 10 seconds. During the test, 190 temporary saves and 29 application submissions were achieved per minute on one web and one application server with two clones. CPU on the web server, HPN8, was 71% while the application server (HPN3) was 70% utilization. The number of concurrent users will be increased to determine the application performs in following cycles. Memory utilization levels for HPN3 and HPN8 were 54% and 50%, respectively.

Conclusions-Cycle 10

- Student Access business process created 18KB session during the test. The database table that was used for the test had 16KB block size. As a result, the 18KB session caused row chaining.
- Determine the cause of DB2 dead lock issue. The application timeout interval was shorter than the DB2 lock wait timeout interval of 120 seconds.
- Demonstrated performance gain from using a HP N-class machine as the web server.
- Confirmed performance gain from using a HP N-class machine as the web server. By adding the second application server, an increase in throughput was realized with less frequent garbage collection.
- Max client setting on a given web server, if increased from 1024 to 2048, throughput will increase and response time decreases.

- Test was aborted due to exceptions returned from a network communication error.

4.11 Performance Test – Cycle 11

Test 1

- FAFSA 7.0 Performance Test Cycle 11 was executed on November 12, 2002.
- Hardware & configuration:
  - Web server – HPN8 (8-way, 750 MHz, 8 GB memory. Max client set to 2048)
  - App server – HPN3 (8-way, 360 MHz, 8 GB memory)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: Check Status
- Goals: The goal of this test was to investigate potential code, environment, or script issues for Check Status business process.

Summary of Results:
Check Status ran smoothly but the script was creating a large number of session records (3215 session records) in 30 minutes with 500 users. JSP directives should be added to FAFSA application code to resolve this issue.

Test 2

- FAFSA 7.0 Performance Test Cycle 11 was executed on November 12, 2002.
- Hardware & configuration:
  - Web server – HPN8 (8-way, 750 MHz, 8 GB memory. Max client set to 2048)
    - MaxSpareServers changed from 100 to 10, MinSapreServers from 10 to 5 and MaxRequestPerChild from 10,000 to 0.
  - App server – HPN3 (8-way, 360 MHz, 8 GB memory)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: FAFSA TempSave-Restore, Fill Out FAFSA 6.0, Fill Out FAFSA 7.0, and FAFSA Corrections 7.0
- Goals: The goal of this test was to determine the number of applications submitted and temporary saves by one web server and three application server clones with 1GB heap size.

Summary of Results:
Performance gain in terms of hits per second, throughput and response time were observed during the first 1000 concurrent users. At 1000 users level, hits per second were 2613 while response time was 8.62 seconds. There were 29 applications submitted per minute and 190 temporary saves per minute. After holding at 1000 users for an extended time, the load was increased up to 1500 users and performance degradation started to show. At 1500 concurrent level, hits per second was at 2122 with 24.8 seconds response time. Application submission was at 11 per minute and temporary saves was at 190 per minute.

Test 3

- FAFSA 7.0 Performance Test Cycle 11 was executed on November 12, 2002.
- Hardware & configuration:
  - Web server – HPN8 (8-way, 750 MHz, 8 GB memory. Max client set to 2048)
  - App server – HPN3 (8-way, 360 MHz, 8 GB memory)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: FAFSA TempSave-Restore, Fill Out FAFSA 6.0, Fill Out FAFSA 7.0, and FAFSA Corrections 7.0
- Goals: The goal of this test was to determine the number of applications submitted and temporary saves by one web server and three application server clones with 512MB heap size.

Summary of Results:
This test led to the determination of the optimum application heap size to be 1024MB. Frequent garbage collection was observed in the test when the heap size was lowered to 512MB. Application submissions were 17 per minute while temporary saves were at 124 per minute. The test was stopped once the constraint in heap size was realized.

Test 4

- FAFSA 7.0 Performance Test Cycle 11 was executed on November 13, 2002.
- Hardware & configuration:
  - Web server – HPN8 (8-way, 750 MHz, 8 GB memory. Max client set to 1024)
  - App server – HPN3 (8-way, 360 MHz, 8 GB memory)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: FAFSA TempSave-Restore, Fill Out FAFSA 6.0, Fill Out FAFSA 7.0, and FAFSA Corrections 7.0.
- Goals: The goal of this test was to determine the number of applications submitted and temporary saves by one web server and two application server clones.

Summary of Results:
This test was not completed due to an error that occurred in CPS mainframe while running the FAFSA Corrections 7.0 script. The following message was in the CPS log:

C7394W11: TRAN W411 CODE .... FROM MSG W11: SQLCODE -811 ON SELECT FROM TCPS2031 (PARAGRAPH 2010), ID = 101010405TE

After some investigation it was found that mainframe code was merged in the performance test environment on the afternoon of 11/12. There were some problems with this new code thus FAFSA Corrections 7.0 script was not run. Once this code is fixed all four business processes will be run once more. The error type was “-811: Result of embedded select statement is a table of more than one row.”

Test 5

- FAFSA 7.0 Performance Test Cycle 11 was executed on November 13, 2002.
- Hardware & configuration:
  - Web server – HPN8 (8-way, 750 MHz, 8 GB memory. Max client set to 1024)
  - App server – HPN3 (8-way, 360 MHz, 8 GB memory)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Infrastructure Test: Network test with FAFSA Home Page 90K File and 6K GIF file
- Goals: The goal of this test was to determine how much throughput would be gained inside the firewall by hitting the FAFSA 7.0 homepage.

Summary of Results:
This test was completed with 1500 users accessing a 90K file on the FAFSA home page with a maximum throughput of 120Mb / sec.

This test was completed with 700 users, accessing the 6K GIF file, with a maximum throughput of 120Mb / sec.

Test 7

- FAFSA 7.0 Performance Test Cycle 11 was executed on November 13, 2002.
- Hardware & configuration:
  - Web server – HPN8 (8-way, 750 MHz, 8 GB memory. Max client set to 1024)
  - App server – HPN3 (8-way, 360 MHz, 8 GB memory)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: FAFSA TempSave-Restore, Fill Out FAFSA 6.0, and Fill Out FAFSA 7.0.
- Goals: The goal of this test was to determine the number of applications submitted and temporary saves by one web server and two application server clones.

Summary of Results:
An out of memory occurred after 30 minutes of running 2000 virtual users. The log file was examined, which indicated that the two clones in this configuration could not support 2000 users. The script percentage distribution will be used to re-run the test with three clones on the application server.

Test statistics for percentage utilization of CPU and Memory were 21% and 56% for HPN3 (app server), respectively. For HPN8 (web server) the percentage utilization of CPU and Memory were 29% and 44%. There were 1913 sessions and garbage collection ranged between 8 and 11 seconds. Application submission was 12 per minute and 192 temporary saves were observed.

Test 8

- FAFSA 7.0 Performance Test Cycle 11 was executed on November 13, 2002.
- Hardware & configuration:

- Web server – HPN8 (8-way, 750 MHz, 8 GB memory. Max client set to 1024)
- App server – HPN3 (8-way, 360 MHz, 8 GB memory)
- Database – DB2 on CICS mainframe environment
- Messaging – MQ handled traffic between app server and DB2

- Business Process: FAFSA TempSave-Restore, Fill Out FAFSA 6.0, and Fill Out FAFSA 7.0.
- Goals: The goal of this test was to determine the number of applications submitted and temporary saves by one web server and three application server clones.

Summary of Results:
This test with the following distribution of virtual users will need to be run again: FAFSA TempSave-Restore 70%, Fill Out a FAFSA 7.0 22%, and Fill Out a FAFSA 6.0 8%. This test produced results that will be used for the purpose of capacity planning for FAFSA at peak. The goal for the next test would be to reduce the number of temporary saves per minute from 233 to 140 and increase the number of application submissions from 14 to 25.

Test statistics for percentage utilization of CPU and Memory were 42% and 31% for HPN3 (app server), respectively. For HPN8 (web server) the percentage utilization of CPU and Memory were 60% and 46%. There were 1940 sessions.

Conclusions-Cycle 11

- Check Status business process created large number of session records.
- The following changes to the web server enhanced the server's performance:
  - MaxSpareServers changed from 100 to 10.
  - MinSapreServers from 10 to 5.
  - MaxRequestPerChild from 10,000 to 0.
- Optimum application heap size is 1024MB with three server clones.
- Mainframe code issue was discovered in Test 4.
- The maximum throughput for the 90K FAFSA Homepage is 120Mb / sec.
- The maximum throughput for a FAFSA Pull 6K GIF file is 120Mb / sec.

4.12 Performance Test – Cycle 12

Test 1

- FAFSA 7.0 Performance Test Cycle 12 was executed on November 14, 2002.
- Hardware & configuration:
  - Web server – HPN3 (8-way, 360 MHz, 8 GB memory. Max client set to 2048)
  - App server – HPN8 (8-way, 750 MHz, 8 GB memory)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: FAFSA TempSave-Restore, Fill Out FAFSA 6.0, Fill Out FAFSA 7.0, and FAFSA Corrections 7.0.
- Goals: The goal of this test was to determine the number of applications submitted and temporary saves by one web server and three application server clones with 1 GB heap size each.

Summary of Results:
A 2000-user load was applied to the system. In the first 30 minutes, 21 application submissions and 251 temporary saves per minute were achieved. Minutes later, connection time out to the web site occurred. As a result, LoadRunner received "out of buffer size" error. The test was terminated prematurely. Following suggestions from Mercury Interactive, Windows Registry values were changed on the Load Generator boxes to resolve the issue.

The test was restarted after the Windows Registry change. 2000 users were loaded once more. 19 application submissions and 191 temporary saves per minute were reached while rerunning the test. Web server (HPN3) had nearly 100% CPU utilization at 2000 concurrent user load.

Test 2

- FAFSA 7.0 Performance Test Cycle 12 was executed on November 14, 2002.
- Hardware & configuration:
  - Web server – HPN3 (8-way, 360 MHz, 8 GB memory. Max client set to 2048)
  - App server – HPN8 (8-way, 750 MHz, 8 GB memory. Web thread count set to 200)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: FAFSA TempSave-Restore, Fill Out FAFSA 6.0, Fill Out FAFSA 7.0, and FAFSA Corrections 7.0.
- Goals: The goal of this test was to determine the number of applications submitted and temporary saves by one web server and three application server clones with 1 GB heap size each.

Summary of Results:
For this test, HPN3 was used as the application server and HPN8 as the web server. The test was able to maintain 2000 concurrent users load for one hour. 26 application submissions and 226 temporary saves per minute were achieved. CPU utilization on HPN3 and HPN8 were 55% and 68% while memory utilization were 68% and 66%, respectively. The test showed that with one web server and one application server with three clones, the application can support 20% of peak FAFSA submissions and temporary saves.

Conclusions-Cycle 12

- This test demonstrated that LoadRunner Generator machines required a configuration update to the Windows Registry.
- Provided capacity planning projections for an application server with three clones.

4.13 OC3 Link Test

Test 1

- Mercury Interactive executedOC3 Link test on November 17, 2002.

- Goals: The goal of this test was to determine whether CSC's upgrade of its network connection link to the Internet would handle peak load usage. The new network has capacity to handle 155 Mb per sec.

Conclusion – OC3 Link Test

- FAFSA home page was able to achieve 120 Mb per second throughput.
- The test GIF file was able achieve 147 Mb per second throughput.

4.14 Performance Test – Cycle 13

Test 1

- FAFSA 7.0 Performance Test Cycle 13 was executed on November 19, 2002.
- Hardware & configuration:
  - Web server – HPN3 (8-way, 360 MHz, 8 GB memory. Max client set to 2048)
  - App server – HPN8 (8-way, 750 MHz, 8 GB memory. Web thread count set to 200)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: FAFSA TempSave-Restore, Fill Out FAFSA 6.0, Fill Out FAFSA 7.0, and FAFSA Corrections 7.0.
- Goals: The goal of this test was to determine the number of applications submitted and temporary saves by one web server and two application server clones.

Summary of Results:
For this test, 2000 concurrent users were loaded during the test. At this concurrency level, the response time was averaging about 17 seconds while the throughput was at 7.2 Mb per second. The CPU and Memory on HPN3 were 95% and 63% utilized and on HPN8 were 21% and 50% utilized. 13 application submissions and 18 temporary saves per minute were achieved. During the test, it was found that the 200-thread count was not performing as well as the previous 75-thread count. 200 thread count led to instability in session database connections. It was decided to switch back to 75-thread count for the subsequent testing.

Test 2

- FAFSA 7.0 Performance Test Cycle 13 was executed on November 19, 2002.
- Hardware & configuration:
  - Web server – HPN3 (8-way, 360 MHz, 8 GB memory. Max client set to 1024)
  - App server – HPN8 (8-way, 750 MHz, 8 GB memory. Web thread count set to 75)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: FAFSA TempSave-Restore, Fill Out FAFSA 6.0, Fill Out FAFSA 7.0, and FAFSA Corrections 7.0.
- Goals: The goal of this test was to determine the number of applications submitted and temporary saves by one web server and two application server clones.

Summary of Results:
For this test, 2000 concurrent users were loaded into the system. The response time at the 2000 users level was at 41 seconds and the throughput was at 9.6 Mb per second. CPU and Memory utilization for HPN3 were 99% and 63%, 25% and 58% for HPN8, respectively. 23 temporary saves per minute and 25 submissions per minute were reached during the test. At the start of the test, CICS was disconnected from DB2. The disconnect command manifested itself in the application error log as "AEY9 transaction abends". The cause was due to an error whereby a disconnect command was issued accidentally by CSC.

Test 3

- FAFSA 7.0 Performance Test Cycle 13 was executed on November 19, 2002.
- Hardware & configuration:
  - Web server – HPL14 & HPL17 (8-way, 440 MHz, 8 GB memory. Max client set to 1024)
  - App server – HPN8 (8-way, 750 MHz, 8 GB memory. Web thread count set to 75)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: FAFSA TempSave-Restore, Fill Out FAFSA 6.0, Fill Out FAFSA 7.0, and FAFSA Corrections 7.0.
- Goals: The goal of this test was to determine the number of applications submitted and temporary saves by one web server and two application server clones

Summary of Results:
For this test, 2000 users were loaded into the system. In order to put more load on the application server to get the number of temporary saves and application submissions, two web servers were used for the test. CPU and Memory were 95% and 50% on HPL14, 80% and 61% on HPL17 and 80% and 63% on HPN8. 190 temporary saves per minute were achieved before the end of the test. This test was cut short time due to constraints of the allowable testing window.

Test 4

- FAFSA 7.0 Performance Test Cycle 13 was executed on November 20, 2002.
- Hardware & configuration:
  - Web server – HPL14 & HPL17 (8-way, 440 MHz, 8 GB memory. Max client set to 1024)
  - App server – HPN8 (8-way, 750 MHz, 8 GB memory. Web thread count set to 75)
  - Database – DB2 on CICS mainframe environment
  - Messaging – Shadow Direct handled traffic between app server and DB2
- Business Process: School Code Search
- Goals: The goal of this test was to execute School Code Search script and determine Shadow Direct's performance under load.

Summary of Results:
This test executed with 2000 users (increasing 500 users at a time). At the 500 users level, CICS was processing 240 transactions per minute and using 6% CPU utilization on the mainframe. The DB2 service time (response time) was 0.010 seconds. The number of get pages per commit was 488. In previous tests using MQ, the number of get pages was below 100. This means that the DB2 was working harder to do the same work with Shadow Direct compared to the tests that were executed with MQ. The mainframe DBA is investigating this matter. With 1000 users, CICS was processing 440 transactions / minute with 0.014 seconds DB2 service time and 10% CPU utilization on the mainframe. With 1500 users, CICS was processing 700 transactions per minute with 0.016 seconds DB2 service time and 18% CPU utilization on the mainframe. With 2000 users, CICS was processing 900 transactions per minute with 0.015 seconds DB2 service time and 23% CPU utilization on the mainframe. Shadow Direct was able to handle the load that has been tested to date.

Test 5

- FAFSA 7.0 Performance Test Cycle 13 was executed on November 20, 2002.
- Hardware & configuration:
  - Web server – HPL14 & HPL17 (8-way, 440 MHz, 8 GB memory. Max client set to 1024)
  - App server – HPN8 (8-way, 750 MHz, 8 GB memory. Web thread count set to 75)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: Check Status
- Goals: The goal of this test was to verify whether sessions were being invalidated upon user exit of Check Status business process.

Summary of Results:
This test was executed with 200 users. It was observed that there were about 500 sessions in the session database and all their sizes were 0. Upon completion of the business process, the LoadRunner script accurately selects the exit button, an action which should invalidate the session records but does not. After investigating this issue it was found that the sessions were set to NULL (i.e., not performing full invalidation) upon exiting the business process. Full session invalidation needs to be in place for this business process. This test will be repeated once the full session invalidation code is in place.

Conclusions-Cycle 13

- FAFSA Performance Test 13 FAFSA TempSave-Restore, Fill Out a FAFSA 7.0, FAFSA Corrections 7.0, and Fill Out a FAFSA 6.0 to determine the number of application submissions and temporary saves that can be achieved by two application server clones is in progress. Several issues occurred during this cycle, the investigation of which is underway.

- School Code Search test to determine the hardware configuration for FAFSA 7.0 was successfully completed.
- Check Status test to verify that sessions were being invalidated was not completed.

4.15 Performance Test – Cycle 14

Test 1

- Mainframe Saturation Test - Cycle 14 - was executed on November 22, 2002.

Protocol

- Close the MQSeries – CICSPTST Bridge.
- Generate a number of messages to the MQ queue.
- Open the MQ – CICSPTST Bridge.

The Test Series

- Test 1: 3,000 “trivial” transactions
- Test 2: 10,000 “trivial” transactions
- Test 3: 3,000 “non-trivial” transactions

The Environment:

CICS

- MAXTASK = 75
- DB2 MAX THREADS = 15

DB2

Test 1: 3,000 “trivial” transactions.

The message queue was primed with 3,000 transactions. The MQ-CICS Bridge was opened and 24 seconds later the message queue was empty. CICS did not reach MAXTASK nor did it reach max DB2 threads. In the meantime, the CPS processor utilization was 100%busy. All SLAs for all workloads on the CPS processor were met. Transaction processing rate was 7,500 per minute, or 125 per second.

Test 2: 10,000 “trivial” transactions

The message queue was primed with 10,000 transactions and the MQ-CICS Bridge was opened. 79 seconds later, the message queue was empty. CICS reached MAXTASK and max DB2 threads, and the CPS processor utilization was 100% busy.

Test 2 accounted for 84% of the processor utilization, while regular production activities accounted for 16% of the processor utilization. All SLAs for all workloads on the CPS processor were met. Transaction processing rate was 7,500 per minute, or 125 per second.

Test 3: 3,000 "non-trivial" transactions

The message queue was primed with 3,000 transactions and the MQ-CICS Bridge was opened. 62 seconds later, the message queue was empty. CICS reached MAXTASK and max DB2 threads. The CPS processor utilization was 90% busy. Test 3 accounted for 70% of the processor utilization, while regular production activities accounted for 20% of the processor utilization. All SLAs for all workloads on the CPS processor were met. Transaction processing rate: 2,880 per minute, or 48 per second.

Assumptions

- The "non-trivial" transaction used in test 3 is representative of the FAFSA 7.0 workload; that is, the "average" FAFSA 7.0 transaction will do a similar number of DB2 calls, consume a similar amount of CPU time, etc.
- The FAFSA 7.0 peak transaction volume (to CICS/DB2) will be approximately 1,700,000 per day, distributed across a 16-hour span. The average arrival rate will be 106,250 per hour, or 1,770 per minute, or 30 per second.
- There is no estimate for peak transaction arrival rate. The transaction service rate observed in test 3 (48 per second at 70% processor busy) is 62% larger than the assumed average service rate.

Conclusions – Mainframe Saturation Test

- The CPS mainframe will need to deliver 96 MIPS to the FAFSA 7.0 application in order to sustain a throughput rate of 30 transactions per second (30/48 times 70% times 219 MIPS). This leaves 123 MIPS available for other work (approximately last month's peak hour demand).
- The CPS mainframe will need to deliver 153 MIPS to the FAFSA 7.0 application in order to sustain a throughput rate of 48 transactions per second (70% times 219 MIPS). This would leave approximately 66 MIPS available for other work, which, depending on the time of day might not be enough.

The CICS configuration needs review:

- Fewer DB2 threads were available than were transactions wanting to use DB2. It is not known how increasing the number of DB2 threads will affect transaction throughput. It is expected that increasing the number of threads will increase CPU demand, memory demand, and quite possibly I/O rate
- Transactions were unable to start in CICS due to the MAXTASK limit. Increasing the value this parameter will increase the number of concurrent

transactions. This will increase CPU demand and will also affect the required number of DB2 threads.

- Should the FAFSA 7.0 workload be merged into one of the existing production CICS regions, or should it be placed in its own CICS region. This introduces issues of availability in addition to performance.

The DB2 configuration needs review:

- How will increasing the number of CICS (FAFSA 7.0) threads affect DB2.
- DB2 buffer pool statistics need review to determine how much adjustment to the number of buffers in each of the various pools is required to handle FAFSA 7.0.
- What'll be the performance impact of changing CICS's ACCOUNTREC parameter from "NONE" to "TASK". The parameter change causes DB2 to cut one accounting record per CICS task instead of one accounting record per thread termination. There will be more DB2 accounting records, greater statistical detail, easier quantification of the overheads associated with each of the various CICS tasks, but at the cost imposed by the data collection process.

Test 2

- FAFSA 7.0 Performance Test Cycle 14 was executed on November 26, 2002.
- Hardware & configuration:
  - Web server – HPN13 (8-way, 750 MHz, 8 GB memory. Max client set to 2048)
  - App server – HPN8 (8-way, 750 MHz, 8 GB memory. Web thread count set to 25, 4 clones, 512 MB heap size each)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: FAFSA TempSave-Restore, Fill Out a FAFSA 6.0, Fill Out a FAFSA 7.0, FAFSA Corrections 7.0
- Goals: The goal of this test cycle was to run 2000 users to determine the number of application submissions and temporary saves that could be achieved by using two N-class machines (8x750 MHz) -- one as a web server and the other as an application server with 4 clones. Each clone had 512 MB heap size and 25-thread count.

Summary of Results:
2000 LoadRunner users were loaded into the system during the test. The test ran smoothly and the statistics produced as an outcome of this test were useful for the peak capacity planning effort.

At 1,500 users level, CPU utilizations on HPN8 and HPN13 were 25% and 42% while the memory utilizations were at 51% and 47%. 32 applications per minute were submitted.
At 2,000 users level, CPU on HPN8 and HPN13 was 30% and 62% utilized, while memory was at 52% and 52% respectively. At this user lever, 33 applications per minute and 73 temporary saves per minute were achieved. It was also noted that application server memory was running

low at 2,000 users level. The number of data sources was fairly stable at 25 connections. There were 1840 sessions in the session database.

Test 3

- FAFSA 7.0 Performance Test Cycle 14 was executed on November 27, 2002.
- Hardware & configuration:
  - Web server – HPN13 (8-way, 750 MHz, 8 GB memory. Max client set to 2048)
  - App server – HPN8 (8-way, 750 MHz, 8 GB memory. Web thread count set to 25)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: Fill Out a FAFSA 6.0, Fill Out a FAFSA 7.0, FAFSA Corrections 7.0, and FAFSA TempSave-Restore
- Goals: The goal of this test cycle was to run 2000 users to determine the number of application submissions and temporary saves that can be achieved by using two N-class machines (8x750 MHz) -- one as a web server and the other as an application server with 4 clones. Each clone had 512 MB heap size and 25-thread count.

Summary of Results:
2000 concurrent users were loaded into the system for this test. It was discovered that two HPL class servers were sharing the load with HPN13 on the web end. The two HPL class servers were brought down and LoadRunner sustained some errors as a result of failing virtual users. 30 minutes after the errors had occurred, the session database connections had reached a maximum of 100; the web server HPN13 was a bottleneck -- total web connections reached their maximum at 2049 while active connections were 1190. On average, the application submission rate was 30, and the rate of temporary saves was 104 per minute. The CPU utilization on HPN13 fluctuated between 80% and 90%. LoadRunner scripts were being timed out waiting for connections. CICS was processing up to 821 transactions per minute.

At the 2000-user level, CPU utilizations on HPN8 and HPN13 were 52% and 61% while the memory utilizations were at 85% and 51%. 24 applications per minute were submitted. During the ramp down of this test, the following error was observed in the log file.
MQ: CSQV086E MQ abnormal connection 00E50705

MQ could not be restarted because the Restart was hung and it was not possible to connect to the log manager. After some investigation it was found that the MQ abend was caused by a timing issue that could result in a "deadlock" situation between the log archive function and the channel interface function. MQ restarted within seconds of the error message getting its appropriate reply. All necessary information has been collected. CSC is researching the issue with IBM support.

Conclusions-Cycle 14

- The mainframe saturation test produced results which were considered as part of the capacity planning effort.

- Fill Out a FAFSA 6.0, Fill Out a FAFSA 7.0, FAFSA Corrections 7.0, and FAFSA TempSave-Restore scripts were run to determine the number of application submission and temporary saves that can be achieved by 8X750 N class web server and an application server with 4 clones. This test produced useful results for the capacity planning effort.

- In Test 3, an MQ abend occurred which were investigated.

4.16 Performance Test – Cycle 15

Test 1

- FAFSA 7.0 Performance Test Cycle 15 was executed on December 3, 2002.
- Hardware & configuration:
  - Web server – HPN3 (8-way, 360 MHz, 8 GB memory. Max client set to 2048) & HPN13 (8-way, 750 MHz, 8 GB memory. Max client set to 2048)
  - App server – HPN8 (8-way, 750 MHz, 8 GB memory. Web thread count set to 25)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: Fill Out a FAFSA 6.0, Fill Out a FAFSA 7.0, FAFSA Corrections 7.0, and FAFSA TempSave-Restore
- Goals: The goal of this test was to determine the number of application submissions and temporary saves that could be achieved by two web servers and an application server with 4 clones. The goal is to achieve 20% of FAFSA peak transactions.

Summary of Results:

A 2000-user load was applied to the application. At this load level, the system was able to process 45 FAFSA application submissions per minute with 3247 concurrent sessions. The CPU and memory utilizations were: HPN3 - 11% CPU, 40% Memory; HPN13 - 75% CPU, 56% Memory; HPN8 - 49% CPU, 59% Memory.

At this load level, CICS completed 553 transactions per minute. During the test, it was observed in the log file that CICS transaction abend had occurred. Transaction time outs were taken from getting messages from the bridge monitor in the CICS region. In order to resolve this issue, increasing the number of bridge monitors is being considered.

Test 2

- FAFSA 7.0 Performance Test Cycle 15 was executed on December 3, 2002.
- Hardware & configuration:
  - Network Dispatcher - Weights on eNetwork Dispatcher were changed due to load distribution issue.
  - Web server – HPN3 (8-way, 360 MHz, 8 GB memory. Max client set to 2048) & HPN13 (8-way, 750 MHz, 8 GB memory. Max client set to 2048)
  - App server – HPN8 (8-way, 750 MHz, 8 GB memory. Web thread count set to 25. Heap size set to 512 MB)

  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: FAFSA TempSave-Restore, Fill Out a FAFSA 6.0, Fill Out a FAFSA 7.0
- Goals: The goal of this test was to determine the number of application submissions and temporary saves that could be achieved by two web servers and an application server with 4 clones. The goal is to achieve 20% of FAFSA peak transactions.

Summary of Results:
FAFSA Corrections 7.0 script was removed because data used in the script was not processed in the mainframe.

2000 users load was applied to the application. 44 application submissions per minute were achieved with 2181 concurrent sessions. The CPU and memory utilizations were: HPN3 - 15% CPU, 40% Memory; HPN13 - 75% CPU, 54% Memory; HPN8 - 48% CPU, 54% Memory.

CICS performed 618 transactions per minute. CICS transaction abend occurred during the test due to the message time out issue.

Test 3

- FAFSA 7.0 Performance Test Cycle 15 was executed on December 3, 2002.
- Hardware & configuration:
  - Web server – HPN3 (8-way, 360 MHz, 8 GB memory. Max client set to 2048) & HPN13 (8-way, 750 MHz, 8 GB memory. Max client set to 2048)
  - App server – HPN8 (8-way, 750 MHz, 8 GB memory. Web thread count set to 25. Heap size set to 512 MB)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: FAFSA TempSave-Restore, Fill Out a FAFSA 6.0, Fill Out a FAFSA 7.0, and FAFSA Corrections 7.0
- Goals: The goal of the test was to stress CICS and DB2.

Summary of Results:
1000 users load was applied to stress the CICS and DB2. CICS was able to process 770 transactions per minute with 16% CPU utilization. DB2 response time reached as high as 0.021 seconds while threads reached 14. In order to further stress CICS and DB2, School Code Search needed to be substituted. The test was stopped to make the change.

Test 4

- FAFSA 7.0 Performance Test Cycle 15 was executed on December 3, 2002.
- Hardware & configuration:
  - Web server – HPN3 (8-way, 360 MHz, 8 GB memory. Max client set to 2048) & HPN13 (8-way, 750 MHz, 8 GB memory. Max client set to 2048)
  - App server – HPN8 (8-way, 750 MHz, 8 GB memory. Web thread count set to 25. Heap size set to 512 MB)

  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: FAFSA TempSave-Restore, Fill Out a FAFSA 6.0, Fill Out a FAFSA 7.0, and FAFSA Corrections 7.0
- Goals: The goal of the test was to stress CICS and DB2.

Summary of Results:
1,500 users load was applied to stress the CICS and DB2. CICS was able to handle 1841 transactions per minute with 42% CPU utilization. DB2 reached 21 threads during the test; the limit was set to 30. Max tasks for each thread was maximized to its upper limit. CICS abend occurred once more due to message time outs. The test was stopped.

Test 5

- FAFSA 7.0 Performance Test Cycle 15 was executed on December 4, 2002.
- Hardware & configuration:
  - Web server – HPN3 (8-way, 360 MHz, 8 GB memory. Max client set to 1024) & HPN13 (8-way, 750 MHz, 8 GB memory. Max client set to 1024)
  - App server – HPN8 (8-way, 750 MHz, 8 GB memory. Web thread count set to 25. Heap size set to 512 MB)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: FAFSA TempSave-Restore, Fill Out a FAFSA 6.0, Fill Out a FAFSA 7.0, and FAFSA Corrections 7.0
- Goals: The goals of this test cycle were to determine the number of application submissions and temporary saves that can be achieved by two web servers and an application server with 4 clones, and to achieve 20% of FAFSA peak transactions.

Summary of Results:
At the 1500 user level, the test statistics for percentage utilization of CPU and Memory were 56% and 62% for HPN8 (app server), respectively. For HPN3 and HPN13 (web servers) the percentage utilization of CPU and Memory were 90% and 34%, 89% and 33%, respectively. 1500 users load was reached in the test before stopping. The system processed 70 FAFSA 7.0 and 27 FAFSA 6.0 application submissions per minute, 110 temporary saves per minute with 1,458 concurrent sessions.

CICS completed 934 transactions per minute using 15% CPU. DB2 response time was at 0.015 seconds with 8 threads running. During the test, CICS Transaction abend occurred due to message time outs. FAFSA Corrections 7.0 data was not processed in the mainframe and the test received data related errors. In addition, IP Spoofing was not working correctly on the LoadRunner side for load balancing purposes. The test was stopped to perform necessary modifications.

Test 6

- FAFSA 7.0 Performance Test Cycle 15 was executed on December 4, 2002.
- Hardware & configuration:

  - Web server – HPN3 (8-way, 360 MHz, 8 GB memory. Max client set to 1024) & HPN13 (8-way, 750 MHz, 8 GB memory. Max client set to 1024)
  - App server – HPN8 (8-way, 750 MHz, 8 GB memory. Web thread count set to 25. Heap size set to 512 MB)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: FAFSA TempSave-Restore, Fill Out a FAFSA 6.0, Fill Out a FAFSA 7.0, and FAFSA Corrections 7.0
- Goals: The goal of this test cycle was to determine the number of application submissions and temporary saves that could be achieved by two web servers and an application server with 4 clones, and to achieve 20% of FAFSA peak transactions.

Summary of Results:
A 2000 user load was applied to the application. The test statistics for percentage utilization of CPU and Memory were 84% and 63% for HPN8 (app server), respectively. For HPN3 and HPN13 (web servers) the percentage utilization of CPU and Memory were 93% and 42%, 79% and 45%, respectively.

The system processed 85 FAFSA 7.0 and 26 FAFSA 6.0 application submissions per minute, along with 161 temporary saves per minute when 3404 concurrent sessions were in the system. One of the 'IP Spoofed' IP addresses was not working properly and caused numerous errors that resulted in high session count.

CICS processed 919 transactions per minute with 18% CPU utilization. DB2 response time was at 0.014 seconds running 15 threads. The test was ended prematurely due to connection failure between HPN8 and CPS.

Test 7

- FAFSA 7.0 Performance Test Cycle 15 was executed on December 4, 2002.
- Hardware & configuration:
  - Web server – HPL14 & HPL17 (8-way, 440 MHz, 8 GB memory. Max client set to 1024)
  - App server – HPN8 (8-way, 750 MHz, 8 GB memory. Web thread count set to 25. Heap size set to 512 MB, with 4 clones)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: PIN Authentication (Web Services)
- Goals: The goal of this test cycle was to validate resolution of an outstanding issue: Low response time on PIN Web Services Test Harness. The goal was to run 500 virtual users.

Summary of Results:
This test ran with 500 virtual users that accessed the PIN Authentication Web Services test harness. The test was able to find hung threads in the test harness application server on SU35e11. The hung threads on SU35e11 were waiting to read from connection sockets on web

servers HPL14 and HPL17. The number of sockets was maximized on SU35e11, thus overloading the test harness. Both test harness and test environment are currently being investigated.

Test statistics for percentage utilization of CPU and Memory were 10% and 64% for HPN8 (app server), respectively. For HPL14 and HPL17 (web servers) the percentage utilization of CPU and Memory were 4% and 24%, 1% and 23%, respectively.

Conclusions – Cycle 15

- Additional runs will be conducted. CICS transaction abend occurred. Increasing the number of bridge monitors is under consideration.
- Test was stopped to replace scripts in order to place more stress on the mainframe.
- FAFSA Corrections 7.0 data was not processed on the mainframe which generated data related errors. Also, the IP Spoofing function on LoadRunner was not working correctly for load balancing purposes. The test was stopped to carry out necessary modifications.
- PIN Authentication (Web Services) test to validate low response time on PIN Test Harness. This test is in progress.

4.17 Performance Test – Cycle 16

Test 1

- FAFSA 7.0 Performance Test Cycle 16 was executed on December 5, 2002.
- Hardware & configuration:
  - Web server – HPN3 (8-way, 360 MHz, 8 GB memory. Max client set to 1024) & HPN13 (8-way, 750 MHz, 8 GB memory. Max client set to 1024)
  - App server – HPN8 (8-way, 750 MHz, 8 GB memory. Each of the 4 clones had Web thread count set to 25 and heap size set to 512 MB)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: FAFSA TempSave-Restore, Fill Out a FAFSA 7.0, and Fill Out a FAFSA 6.0
- Goals: The goal of this test cycle was to determine the number of application submissions and temporary saves that could be achieved by two web servers and an application server with 4 clones.

Summary of Results:
A 2000 users load was applied to the FAFSA application. The test statistics of percentage utilization of CPU and Memory for the application server (HPN8) were 93% and 58%, respectively; while the web servers HPN3 and HPN13 exhibited the following percentage utilization: 50% CPU and 43% Memory, 10% CPU and 36% Memory, respectively. 3560 concurrent sessions were reached during the test. The configuration was able to achieve 82

FAFSA 7.0 and 21 FAFSA 6.0 application submissions per minute and 190 temporary saves per minute.

CICS processed 1135 transactions per minute with 19% CPU utilization. DB2 response time was at 0.030 seconds and the maximum number of threads was 16.

In the test, a CICS transaction time out occurred. It was found that the timeout occurred at approximately the same time as when MQ log was off loaded to the archive on the mainframe. Off loading happened when the log was full and required human operator intervention. If human operator intervention was not provided in time, time outs would occur. To resolve this issue, "Auto Acknowledgement" needs to be turned on at CPS.

Test 2

- FAFSA 7.0 Performance Test Cycle 16 was executed on December 5, 2002.
- Hardware & configuration:
  - Web server – HPN3 (8-way, 360 MHz, 8 GB memory. Max client set to 1024) & HPN13 (8-way, 750 MHz, 8 GB memory. Max client set to 1024)
  - App server – HPN8 (8-way, 750 MHz, 8 GB memory. Each of the 4 clones had Web thread count set to 25 and heap size set to 512 MB)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: FAFSA TempSave-Restore, Fill Out a FAFSA 7.0, and Fill Out a FAFSA 6.0
- Goals: The goal of this test cycle was to determine the number of application submissions and temporary saves that could be achieved by two web servers and an application server with 4 clones.

Summary of Results:
A total user load of 3000 was applied to the FAFSA application.

At the 2500 users load level, 2575 sessions were reached. The application server (HPN8) statistics for CPU and Memory utilization were: 92% and 62%, respectively. The web servers (HPN3, HPN13) performance statistics for CPU and Memory utilization were: 4% and 26%, 10% and 33% respectively. 87 FAFSA 7.0 and 15 FAFSA 6.0 applications per minute were submitted and 159 temporary saves per minute were achieved. CICS processed 1212 transactions per minute with 20% CPU. DB2 response time was at 0.033 seconds with max threads at 21.

At 3000 users level, 4848 sessions were in the system. The test statistics, at this user level, of percentage utilization of CPU and Memory for the application server (HPN8) were 81% and 61%, respectively; while the web servers HPN3 and HPN13 demonstrated the following percentage utilization of CPU and Memory: 6% and 28%, 10% and 57%, respectively.
87 FAFSA 7.0 and 15 FAFSA 6.0 applications per minute were submitted and 196 temporary saves were achieved. CICS processed 960 transactions per minute with 20% CPU. DB2 response time was at 0.052 seconds with 30 max threads.

Performance degradation was observed at 3000 users load. No additional throughput was gained and exceptions such as "Out of Memory Error" appeared due to large number of sessions in the memory caused by failed virtual users. Database connection error also occurred toward the end of the test:

[ERROR] [Worker#23] FOTWServlet

System Error: AException thrown Thursday, December 5, 2002 12:45 PM AFOTWDatabase.executeProcParms: No connection APPLOOKUP Timeout waiting for free connection.

Test 3

- FAFSA 7.0 Performance Test Cycle 16 was executed on December 6, 2002.
- Hardware & configuration:
  - Web server – HPN3 (8-way, 360 MHz, 8 GB memory. Max client set to 2048) & HPN13 (8-way, 750 MHz, 8 GB memory. Max client set to 2048)
  - App server – HPN8 & HPN15 (8-way, 750 MHz, 8 GB memory. Each of the 8 clones had Web thread count set to 25 and heap size set to 512 MB)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: FAFSA TempSave-Restore, Fill Out a FAFSA 7.0, and Fill Out a FAFSA 6.0
- Goals: The goal of this test cycle was to determine the number of application submissions and temporary saves that can be achieved by two web servers and two application servers. Each application server was configured with 4 clones.

Summary of Results:
3000 user load was applied to the application in the test. The load was applied incrementally in order to observe server behavior under different levels.

At 1500 users level, CPU and Memory utilizations on the servers were: HPN3 - 20% CPU, 23% Memory; HPN13 - 25% CPU, 39% Memory; HPN8 - 30% CPU, 39% Memory; HPN15 - 30% CPU, 29% Memory.

At the 2500 user level, CICS processed 1034 transactions per minute using 19% CPU. CPU and Memory utilizations on the servers were: HPN3 - 35% CPU, 37% Memory; HPN13 - 45% CPU, 48% Memory; HPN8 - 70% CPU, 57% Memory; HPN15 - 70% CPU, 30% Memory.

At the 2800 user level, CICS completed 1929 transactions per minute with 30% CPU. 186 applications per minute were submitted and 282 temporary applications saved. CPU and Memory utilizations on the servers were: HPN3 - 30% CPU, 38% Memory; HPN13 - 25% CPU, 49% Memory; HPN8 - 25% CPU, 51% Memory; HPN15 - 32% CPU, 30% Memory.

CICS completed 1873 transactions per minute with 33% CPU. 186 applications per minute were submitted.

Performance degradation was observed. No additional throughput was gained after pushing the number of users to 2800. Through out the test, CICS transaction time outs were occurring. At the 2800 users level, DB2 thread dead locks began to appear. Max task reached 75. In the application logs, Java Null Pointer exceptions with corrupted time stamp and "no reply received within specified time frame" error message were also observed.

After troubleshooting MQ and CICS it was learned that one of the Connection Pooling configuration properties was not set correctly. The 15 seconds message time out was translated to 1.5 seconds on the Bridge and the messages were timed out prematurely. EAI team is implementing the fix.

Conclusions-Cycle 16

- CICS transaction time out occurred. "Auto Acknowledgement" function needs to be turned on at CPS to obviate the need for human intervention when MQ log is off loaded the mainframe archive.
- CICS transaction time out occurred. Performance degradation at 3000-user level, as well as, "Out of Memory" exceptions were observed.
- Establish capacity planning projections for FAFSA 7.0 peak usage. It was also found that one of the configuration properties on the connection pooling code needed updating.

4.18 Performance Test – Cycle 17

Test 1

- FAFSA 7.0 Performance Test Cycle 17 was executed on December 9, 2002.
- Hardware & configuration:
  - Web server – HPN3 (8-way, 360 MHz, 8 GB memory. Max client set to 2048) & HPN13 (8-way, 750 MHz, 8 GB memory. Max client set to 2048)
  - App server – HPN8 & HPN15 (8-way, 750 MHz, 8 GB memory. Each of the 8 clones had Web thread count set to 25 and heap size set to 512 MB)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: FAFSA TempSave-Restore, Fill Out a FAFSA 7.0, and Fill Out a FAFSA 6.0
- Goals: The goal of this test cycle was to determine the number of application submissions and temporary saves that can be achieved by two web servers and two application servers. Each application server was configured with 4 clones.

Summary of Results:

A 2000 user load was applied to the application. The test statistics for percentage utilization of CPU and Memory were 19% and 56% for HPN8 (app server), respectively. The CPU and

Memory utilization percentages for the second application server HPN15 were 66% and 30%, respectively. As for HPN3 and HPN13 (web servers) the percentage utilization of CPU and Memory were 14% and 36%, 17% and 46%, respectively. The system processed 108 FAFSA 7.0 and 32 FAFSA 6.0 application submissions per minute, along with 220 temporary saves per minute when 4255 concurrent sessions were in the system.

CICS processed 1106 transactions per minute with 21% CPU utilization. DB2 response time was at 0.014 seconds running 15 threads. The test was ended prematurely due to connection failure between HPN8 and CPS.

The test showed that an EAI timeout occurred. Additional errors such as Java Null Pointer exception with corrupted time stamps appeared during the test. Tasks on DB2 thread reached 50 (the maximum) and DB2 threads were also maxed at 15. The backlog in the CICS / DB2 translated into timeouts on the web servers. The test was stopped to increase DB2 max threads to 40.

FAFSA Corrections 7.0 script was removed due to updates in the application code that resulted in the addition of a web page.

Test 2

- FAFSA 7.0 Performance Test Cycle 17 was executed on December 9, 2002.
- Hardware & configuration:
  - Web server – HPN3 (8-way, 360 MHz, 8 GB memory. Max client set to 2048) & HPN13 (8-way, 750 MHz, 8 GB memory. Max client set to 2048)
  - App server – HPN8 & HPN15 (8-way, 750 MHz, 8 GB memory. Each of the 8 clones had Web thread count set to 25 and heap size set to 512 MB)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: FAFSA TempSave-Restore, Fill Out a FAFSA 7.0, and Fill Out a FAFSA 6.0
- Goals: The goal of this test cycle was to determine the number of application submissions and temporary saves that can be achieved by two web servers and two application servers. Each application server was configured with 4 clones.

Summary of Results:

A 2000 user load was applied to the application. The test statistics for percentage utilization of CPU and Memory were 38% and 62% for HPN8 (app server), respectively. The CPU and Memory utilization percentages for the second application server HPN15 were 45% and 33%, respectively. As for HPN3 and HPN13 (web servers) the percentage utilization of CPU and Memory were 18% and 37%, 21% and 50%, respectively. The system processed 25 FAFSA 7.0 and 32 FAFSA 6.0 application submissions per minute, along with 226 temporary saves per minute when 4255 concurrent sessions were in the system.

CICS processed 1000 transactions per minute with 18% CPU utilization. DB2 response time was at 0.071 seconds running 33 active threads. The test was ended prematurely due to connection failure between HPN8 and CPS.

During the test, EAI messages were still timing out on the Bridge. CICS abend with error code CKB5 occurred regularly. Max tasks on DB2 thread was also reached. After extensive troubleshooting, it was concluded that MQ was issuing duplicate message IDs. When the Bridge saw the duplicated message id, it ignored the message, which eventually caused messages to timeout. EAI team is currently working with IBM to resolve this issue.
Following a review of CICS logs, CICS system statistics indicate that there were more tasks than DB2 threads and TCBs, consequently, CICS had been at MAXTASK with 70 tasks waiting. The transaction-level statistics do not indicate that any transactions waited for DB2 threads or for MAXTASK.

FAFSA Corrections 7.0 script was removed due to updates in the application code that resulted in the addition of a web page.

Test 3

- FAFSA 7.0 Performance Test Cycle 17 was executed on December 10, 2002.
- Hardware & configuration:
  - Web server – HPN13 (8-way, 750 MHz, 8 GB memory. Max client set to 2048)
  - App server – HPN8 (8-way, 750 MHz, 8 GB memory. Each of the 4 clones had Web thread count set to 25 and heap size set to 512 MB)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: FAFSA TempSave-Restore, Fill Out a FAFSA 7.0, and Fill Out a FAFSA 6.0
- Goals: The goal of the test was to determine the number of application submissions and temporary saves that can be achieved by one web /one app server configuration.

Summary of Results:
2000 users were loaded into the system for the purpose of this test.

At the 1,500 users level, the server utilizations were: HPN13 - 19% CPU, 40% Memory; HPN8 - 81% CPU, 57% Memory.

CICS completed 1161 transactions per minute using 26% CPU. 96 FAFSA 7.0 application submissions, 26 FAFSA 6.0 application submissions and 170 temporary saves per minute were achieved.

At the 2000 users level, the server utilizations were: HPN13 - 34% CPU, 59% Memory; HPN8: 89% CPU, 57% Memory.

CICS did 1280 transactions per minute using 25% CPU. 104 FAFSA 7.0 application submissions,

21 FAFSA 6.0 application submissions and 189 temporary saves per minute were achieved.

At 2000 users level, minimal throughput was gained from adding the additional 500 users. After running a utility tool provided by IBM that re-synchs MQ message index, no message expiration was observed.

Test 4

- FAFSA 7.0 Performance Test Cycle 17 was executed on December 10, 2002.
- Hardware & configuration:
  - Web server – HPN3 (8-way, 360 MHz, 8 GB memory. Max client set to 2048) & HPN13 (8-way, 750 MHz, 8 GB memory. Max client set to 2048)
  - App server – HPN8 & HPN15 (8-way, 750 MHz, 8 GB memory. Each of the 8 clones had Web thread count set to 25 and heap size set to 512 MB)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: FAFSA TempSave-Restore, Fill Out a FAFSA 7.0, and Fill Out a FAFSA 6.0
- Goals: The goal of this test cycle was to determine the number of application submissions and temporary saves that could be achieved by two web servers and two application servers. Each application server was configured with 4 clones.

Summary of Results:
3000 users were loaded in the test. At the 2,500 users level, the server utilizations were: HPN3 - 43% CPU, 44% Memory; HPN13 - 18% CPU, 39% Memory; HPN8 - 66% CPU, 62% Memory HPN15: 53% CPU, 30% Memory.

CICS completed 1914 transactions per minute using 30% CPU. 135 FAFSA 7.0 application submissions, 30 FAFSA 6.0 application submissions and 266 temporary saves per minute were achieved.

At 3000 users level, the server utilizations were: HPN3 - 41% CPU, 44% Memory
HPN13 - 24% CPU, 44% Memory; HPN8 - 71% CPU, 62% Memory; HPN15 -59% CPU, 30% Memory.

CICS processed 2138 transactions per minute using 37% CPU. 165 FAFSA 7.0 application submissions, 28 FAFSA 6.0 application submissions and 315 temporary saves per minute were achieved.

At 3000 users level, CICS reached max tasks on DB2 threads. DB2 started to complete for system resource and the system placed a lock on the threads. Time outs on threads occurred as a direct result. After this point, transactions started to back log all the way to the web servers and eventually, request time outs started to appear on LoadRunner. DB2 administrators are currently looking into this issue.

Test 5

- FAFSA 7.0 Performance Test Cycle 17 was executed on December 11, 2002.
- Hardware & configuration:
  - Web server – HPN3 (8-way, 360 MHz, 8 GB memory. Max client set to 1024) & HPN13 (8-way, 750 MHz, 8 GB memory. Max client set to 1024)
  - App server – HPN8 & HPN15 (8-way, 750 MHz, 8 GB memory. Each of the 8 clones had Web thread count set to 25 and heap size set to 512 MB)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: FAFSA TempSave-Restore, Fill Out a FAFSA 7.0, and Fill Out a FAFSA 6.0
- Goals: The goal of this test cycle was to run the FAFSA TempSave-Restore, Fill Out a FAFSA 7.0, and Fill Out a FAFSA 6.0 scripts to determine the number of application submission and temporary saves that can be achieved by 2 N-class web servers and 2 N-class app servers with 4 clones on each app server.

Summary of Results:
In total, a 3000-user load was applied to the application.

At the 2800 user level, the test produced the following statistics for percentage utilization of CPU and Memory for the application servers (HPN8, HPN15): 68% and 57% for HPN8, and 64% and 30% for HPN15, respectively. As for the web servers HPN3 and HPN13, the percentage utilization of CPU and Memory were 56% and 40%, 28% and 50%, respectively. The number of application submissions for FAFSA 7.0 was 143 and 30 for FAFSA 6.0, while temporary saves numbered 358 per minute. CICS processed 2328 transactions per minute using 38% CPU. A maximum of 20 DB2 threads were utilized at the 2800 user level.

At 3000 users level the hardware configuration processed 172 FAFSA 7.0 applications, 32 FAFSA 6.0 and 376 temporary saves per minute. The server utilization statistics for the application servers were: 72% CPU, 58% Memory for HPN8, and 88% CPU, and 33% Memory for HPN15. The web server utilization exhibited these statistics: 59% CPU, 56% Memory for HPN3, 30% CPU, and 53% Memory for HPN13. CICS performed 2338 transactions per minute using 41% CPU. 18 DB2 threads were utilized at the 3000 users level.

At 3000 users level, CICS started to wait for DB2 threads. The wait eventually caused EAI “15 seconds no response” timeouts. Research was carried out and it was learned that the symptoms experienced in the performance environment occurred in test environment as well. A patch was applied to the test environment that resolved the problem. The test was stopped in order to apply the patch in the performance environment.

Test 6

- FAFSA 7.0 Performance Test Cycle 17 was executed on December 11, 2002.
- Hardware & configuration:
  - Web server – HPN3 (8-way, 360 MHz, 8 GB memory. Max client set to 2048) & HPN13 (8-way, 750 MHz, 8 GB memory. Max client set to 2048)

  - App server – HPN8 & HPN15 (8-way, 750 MHz, 8 GB memory. Each of the 8 clones had Web thread count set to 25 and heap size set to 512 MB)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: FAFSA TempSave-Restore, Fill Out a FAFSA 7.0, Fill Out a FAFSA 6.0, PIN Registration, and PIN Authentication through HTML
- Goals: The goal of this test cycle was to determine the number of application submissions and temporary saves that can be achieved by two web servers and two application servers. Each application server was configured with 4 clones.

Summary of Results:
In total, a 3000-user load was applied to the application. A few configuration issues occurred that resulted in failures. No timeout on EAI message responses was observed. The test was ended early due to testing time frame. The same test will be rerun at the next cycle.

Conclusions-Cycle 17

- The test was stopped due to a connection failure between HPN8 and CPS. EAI errors occurred during the test, and errors such as Java Null Pointer exception with corrupted time stamp appeared. This test is in progress.
- Established capacity planning projections for FAFSA 7.0 peak under a one web and one app server environment. No CICS transaction time out was observed after the synching up of MQ message index.
- Established capacity planning projections for FAFSA 7.0 peak. CICS max tasks and DB2 max threads reached.
- The test was stopped to apply a patch in the performance test environment for DB2. This test is in progress.
- The test configuration issue arose which resulted in virtual user failure. The test was stopped due to time constraints. No EAI timeouts were observed. This test is in progress.

4.19 Performance Test – Cycle 18

Test 1

- FAFSA 7.0 Performance Test Cycle 18 was executed on December 12, 2002.
- Hardware & configuration:
  - Web server – HPN3 (8-way, 360 MHz, 8 GB memory. Max client set to 2048) & HPN13 (8-way, 750 MHz, 8 GB memory. Max client set to 2048)
  - App server – HPN8 & HPN15 (8-way, 750 MHz, 8 GB memory. Each of the 8 clones had Web thread count set to 25 and heap size set to 512 MB)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: FAFSA TempSave-Restore, Fill Out a FAFSA 7.0, Fill Out a FAFSA 6.0, PIN Registration, and PIN Authentication through HTML
- Goals: The goal of this test was to verify that the DB2 patch that was applied eliminated the DB2 issue. In Cycle 10 test on 11/07/2002, DB2 deadlocks were

observed. A patch was applied and this test will verify that patch has resolved the deadlock issue.

Summary of Results:
A 3000-user load was placed on the application. At this user level, 173 FAFSA 7.0 application submissions, 31 FAFSA 6.0 application submissions and 376 temporary saves per minute were achieved. The server utilizations were at: HPN3 - 63% CPU, 63% Memory; HPN13 - 30% CPU, 53% Memory; HPN8 - 70% CPU, 63% Memory; HPN15 - 73% CPU, 34% Memory.

CICS processed 2357 transactions per minute using 42% CPU. 22 DB2 threads were utilized. The test showed the patch applied to the DB2 reduced the thread waiting time from 11 seconds to less than 1 second.

Out of memory exception was taken on one of eight clones toward the end of the test. A large number of users executing FAFSA Corrections 7.0 failed due to data issue and caused an unusual amount of sessions to be stored in memory. Due to the out of memory errors that were sustained, this patch needs to be retested.

Test 2

- FAFSA 7.0 Performance Test Cycle 18 was executed on December 12, 2002.
- Hardware & configuration:
  - Web server – HPN3 (8-way, 360 MHz, 8 GB memory. Max client set to 2048) & HPN13 (8-way, 750 MHz, 8 GB memory. Max client set to 2048)
  - App server – HPN8 & HPN15 (8-way, 750 MHz, 8 GB memory. Each of the 8 clones had Web thread count set to 25 and heap size set to 512 MB)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: FAFSA TempSave-Restore, Fill Out a FAFSA 7.0, Fill Out a FAFSA 6.0, PIN Registration, and PIN Authentication through HTML
- Goals: The goal of this test was to verify that FAFSA business processes and PIN business processes could function in a shared environment without causing any performance degradations. During the test, a separate load was being placed on CPS by IDC.

Summary of Results:
A 3000-user load was applied to the application. At this load level, 119 FAFSA 7.0 application submissions, 18 FAFSA 6.0 application submissions and 268 temporary saves per minute were achieved. The server utilizations were: HPN3 - 63% CPU, 63% Memory; HPN13 - 30% CPU, 53% Memory; HPN8 - 70% CPU, 63% Memory; HPN15 - 73% CPU, 34% Memory.

CICS processed 1837 transactions per minute of which, 64 transactions came from IDC batch job. CICS used 36% CPU. DB2 threads peaked at 39.

At 3000 users level, application submission was at 153 applications and PIN authentication was at 300 requests per minute. These two numbers both exceeded the estimated peak traffic.

EAI messages timeouts occurred and it was found that duplicated messages were sent to the mainframe (SyncQ). EAI is currently troubleshooting this issue.

Test 3

- FAFSA 7.0 Performance Test Cycle 18 was executed on December 13, 2002.
- Hardware & configuration:
  - Web server – HPN13 (8-way, 750 MHz, 8 GB memory. Max client set to 2048)
  - App server – HPN8 (8-way, 750 MHz, 8 GB memory. Each of the 2 clones had Web thread count set to 40 and heap size set to 1 GB)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: FAFSA TempSave-Restore, Fill Out a FAFSA 7.0, Fill Out a FAFSA 6.0, and FAFSA Renewal
- Goals: The goal of this test was to carry out performance tuning on the WebSphere Application Server (WAS). A one web server and one application server configuration was used for this test. The application server had two clones each with a maximum of 40-thread count and 1 GB heap size. This tuning would determine the number of application submissions, temporary saves, and concurrent users the given configuration can support.

Summary of Results:
A 2000-user load was applied to the application. 73 FAFSA 7.0 application submissions and 20 FAFSA 6.0 application submissions per minute were achieved. The server utilizations were: HPN13 - 16% CPU, 42% Memory; HPN8 - 80% CPU, 54% Memory.

CICS completed 1098 transactions per minute using 18% CPU. 12 DB2 threads were used. During the test, FAFSA 6.0 temporary save table filled up and was cleared. DB2 transactions abend also occurred. EAI and CSC are working with IBM to resolve the transaction abend.

Test 4

- FAFSA 7.0 Performance Test Cycle 18 was executed on December 17, 2002.
- Hardware & configuration:
  - Web server – HPN3 (8-way, 360MHz, 8 GB memory. Max client set to 2048)
  - App server – HPN8 (8-way, 750 MHz, 8 GB memory. Each of the 2 clones had Web thread count set to 40 and heap size set to 1 GB)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: Check Status
- Goals: The goal of this test was to validate resolution of Check Status business process session invalidation.

Summary of Results:
500 users were loaded during the test. It was verified that sessions created by Check Status business process were invalidated instead of nulled.

Test 5

- FAFSA 7.0 Performance Test Cycle 18 was executed on December 17, 2002.
- Hardware & configuration:
  - Web server – HPN3 (8-way, 360MHz, 8 GB memory. Max client set to 1024), HPL17 (8-way, 440 MHz, 8 GB memory. Max client set to 2048)
  - App server – HPN8 (8-way, 750 MHz, 8 GB memory. Each of the 2 clones had thread count set to 40 and heap size set to 1 GB)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: FAFSA TempSave-Restore, Fill Out a FAFSA 7.0, FAFSA Corrections 7.0, and Fill Out a FAFSA 6.0
- Goals: The goal of this web server fail over test was to determine the number of users affected when a web server fails over.

3000 users were generated and loaded into the system. At the 3000-user level, 98 FAFSA 7.0 applications, 15 FAFSA 6.0 applications, 191 FAFSA 7.0 temp saves and 15 FAFSA 6.0 temp saves per minute were achieved. The server utilizations were: HPN3 - 75% CPU, 52% Memory; HPN8 - 77% CPU, 50% Memory.

CICS processed 1283 transactions per minute using 14% CPU. 11 DB2 threads were reached.

One of the web servers (HPL17) was stopped during the test for observing the effect on web server fail over. Network Dispatcher was able to recognize the down server and distribute load to the other server. 116 users on HPL17 were dropped due to the simulated server crash. It was expected that some users would be dropped.

Test 6

- FAFSA 7.0 Performance Test Cycle 18 was executed on December 17, 2002.
- Hardware & configuration:
  - Web server – HPN3 (8-way, 360MHz, 8 GB memory. Max client set to 2048)
  - App server – HPN8 (8-way, 750 MHz, 8 GB memory. Each of the 2 clones had Web thread count set to 40 and heap size set to 1 GB)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: FAFSA TempSave-Restore, Fill Out a FAFSA 7.0, FAFSA Corrections 7.0, and Fill Out a FAFSA 6.0
- Goals: The goal of this test was to determine availability and the number of users affected when Network Dispatcher fails over.

Summary of Results:
3000 users were generated and loaded. At 3000 users level, 112 FAFSA 7.0 applications, 22 FAFSA 6.0 applications and 222 temporary saves per minute were achieved. The server utilizations were: HPN3 - 75% CPU, 55% Memory; HPN8 - 70% CPU, 51% Memory.

CICS processed 1399 transactions per minute using 17% CPU. 11 DB2 threads were reached.

The primary Network Dispatcher was stopped during the test. The secondary ND server was able to pick up requests and distribute load to the web server. 623 users were affected during fail over

Conclusions-Cycle 18

- The test errors occurred as a result of the FAFSA Corrections 7.0 script, the DB2 patch that was applied to correct the DB2 deadlock issue will require retesting. This test is in progress.
- During this test a separate load was placed on CPS by IDC. The number of application submissions and PIN authentication exceeded estimated figures for peak. EAI message timeouts occurred. This matter is being investigated by EAI.
- WAS tuning is in-progress. DB2 transaction abend occurred.
- Resolution of session invalidation issue of sessions created by the Check Status business is confirmed.
- Network Dispatcher (ND) recognized that the web server was down and distributed the load to the other web server in the configuration. An expected, minor loss of users occurred.
- Back up Network Dispatcher server was able to pick up requests after primary Network Dispatcher was decommissioned.

4.20 Performance Test – Cycle 19

Test 1

- FAFSA 7.0 Performance Test Cycle 19 was executed on December 18, 2002.
- Hardware & configuration:
  - Web server – HPN3 (8-way, 360MHz, 8 GB memory. Max client set to 2048)
  - App server – HPN8 (8-way, 750 MHz, 8 GB memory. Each of the 2 clones had Web thread count set to 40 and heap size set to 1 GB)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: FAFSA Corrections 7.0
- Goals: The goal of this test was to validate resolution on a DB2 batch processor issue, which occurred when CPS Compute ran concurrently with FAFSA Corrections 7.0 to stress DB2. A one web server and one application server configuration was used for this test.

Summary of Results:

500 users were loaded during the test. Three sets of batch jobs were loaded and computed. One DB2 abend occurred due to test data loading issue. This data loading issue happened when the system attempted to do a correction on an application that had not been loaded. In real world scenario, this data loading issue will not happen. One EAI transaction time out was logged during the test.

Test 2

- FAFSA 7.0 Performance Test Cycle 19 was executed on December 18, 2002.
- Hardware & configuration:
  - Web server – HPN3 (8-way, 360MHz, 8 GB memory. Max client set to 2048)
  - App server – HPN8 (8-way, 750 MHz, 8 GB memory. Each of the 2 clones had Web thread count set to 40 and heap size set to 1 GB)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: FAFSA PIN Authentication
- Goals: The goal of this test was to validate the FAFSA PIN capacity planning numbers with the current configuration. A one web server and one application server configuration was used for this test.

Summary of Results:
1500 users were loaded during the test. The configuration was able to process 645 transactions per minute using 6% CPU and 41% Memory on HPN8. Several transaction failures occurred during the test. The failures were due to data used in the test.

Test 3

- FAFSA 7.0 Performance Test Cycle 19 was executed on December 18, 2002.
- Hardware & configuration:
  - Web server – HPN3 (8-way, 360MHz, 8 GB memory. Max client set to 2048)
  - App server – HPN8 (8-way, 750 MHz, 8 GB memory. Each of the 2 clones had Web thread count set to 40 and heap size set to 1 GB)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: FAFSA TempSave-Restore, Fill Out a FAFSA 7.0, Fill Out a FAFSA 6.0, and FAFSA Corrections 7.0
- Goals: The goal of this test was to perform an application server failover test. A one web server and two application server configuration was used for this test.

Summary of Results:
1000 users were applied to the configuration. During the test, the clone on HPN8 was killed to simulate an application server crash. The Admin Server was able to sense the down clone and restarted it. 90 of 1000 users were dropped when one clone on HPN8 was killed.

Conclusions-Cycle 19

- Validated a DB2 batch processor fix using FAFSA Corrections 7.0 and CPS Load was completed successfully.
- Confirmed the validity of FAFSA PIN capacity numbers using FAFSA PIN Authentication was completed successfully.
- The application server failover test was completed successfully.

4.21 Performance Test – Cycle 20

Test 1

- FAFSA 7.0 Performance Test Cycle 20 was executed on December 19, 2002.
- Hardware & configuration:
  - Web server – HPN3 (8-way, 360MHz, 8 GB memory. Max client set to 2048)
  - App server – HPN8 (8-way, 750 MHz, 8 GB memory. Each of the 2 clones had Web thread count set to 40 and heap size set to 1 GB)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: PIN Authentication (Web Services)
- Goals: The goal of this test was to work on resolution on PIN Web Services slow response issue.

Summary of Results:
500 users were loaded during the test. It was found the test harness was not closing its TCP connections and filled up the firewall's states table. Once the states table was filled, no connections were going through. A Pearson developer indicated that SOAP did not provide a mechanism to close connections. In the test environment, the states table can hold 25,000 records with a 2-hour time out setting. In production, the table size is set to 200,000 records with 1-hour time out value. Thus, with 500 concurrent users, it took about 30 minutes to fill up the states table in performance test. ITA, Pearson and CSC are working on a resolution. Also, the performance test firewall needs to be updated.

Test 2

- FAFSA 7.0 Performance Test Cycle 20 was executed on December 19, 2002.
- Hardware & configuration:
  - Web server – HPL14 & HPL17 (8-way, 440MHz, 8 GB memory. Max client set to 2048)
  - App server – HPN8 (8-way, 750 MHz, 8 GB memory. Each of the 2 clones had Web thread count set to 40 and heap size set to 1 GB)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: PIN Authentication (Web Services)
- Goals: The goal of this test was to recreate WAS hung process issue for further troubleshooting.

Summary of Results:
1000 users were loaded during the test. The two web servers were kept at 80% CPU usage throughout the test. No hung process was observed after running for an hour.

Conclusions – Cycle 20

- Issue resolution of PIN Web Services slow response is in progress.

- Issue resolution of WAS hung process is in progress.

4.22 Performance Test – Cycle 21

Test 1

- FAFSA 7.0 Performance Test Cycle 21 was executed on December 20, 2002.
- Hardware & configuration:
  - Web server – HPN3 (8-way, 360MHz, 8 GB memory. Max client set to 2048)
  - App server – HPN8 (8-way, 750 MHz, 8 GB memory. Each of the 2 clones had Web thread count set to 40 and heap size set to 1 GB)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: PIN Authentication (Web Services)
- Goals: The goal of this test was to find out if hung process would occur when running at 80% CPU on the web servers.

Summary of Results:
1000 users were loaded during the test. The two web servers were kept at 80% CPU usage throughout the test. No hung process was observed after running the test for an hour.

Conclusion-Cycle 21

- No hung process was observed with respect to the PIN Web Services business process with CPU 80% busy.

4.23 Performance Test – Cycle 22

This cycle was not conducted successfully due to LoadRunner Controller issues that caused LoadRunner to crash repeatedly. A patch was installed and the issue was resolved.

4.24 Performance Test – Cycle 23

Test 1

- FAFSA 7.0 Performance Test Cycle 23 was executed on December 27, 2002.
- Hardware & configuration:
  - Web server – HPN3 (8-way, 360MHz, 8 GB memory. Max client set to 2048) & HPL14 (8-way, 440 MHz, 8 GB memory. Max client set to 2048)
  - App server – HPN8 (8-way, 750 MHz, 8 GB memory. Each of the 2 clones had Web thread count set to 40 and heap size set to 1 GB)
  - Database – DB2 on CICS mainframe environment
  - Messaging – MQ handled traffic between app server and DB2
- Business Process: FAFSA TempSave-Restore, Fill Out a FAFSA 7.0, FAFSA Renewal, Student Access, Check Status, FAA Corrections, PIN Authentication (HTML), and PIN Registration.

- Goals: The goal of this test was to verify that FAFSA business processes and PIN business processes could function in a shared environment without causing any performance degradation.

Summary of Results:
2000 users were loaded during the test. At this load level, the CPU resource usages were: HPN8 - 34% CPU, 54% Memory; HPN3 - 73% CPU, 41% Memory; HPL14 - 46% CPU, 31% Memory. CICS processed 900 transactions per minute using 20% CPU.

The test was able to show that FAFSA and PIN business processes could run in a shared environment without problems.

Test 2

- FAFSA 7.0 Performance Test Cycle 23 was executed on December 27, 2002.
- Hardware & configuration:
  - Web server – HPN3 (8-way, 360MHz, 8 GB memory. Max client set to 2048) & HPL14 (8-way, 440 MHz, 8 GB memory. Max client set to 2048)
  - App server – HPN8 (8-way, 750 MHz, 8 GB memory. Each of the 2 clones had Web thread count set to 40 and heap size set to 1 GB)
  - Database – DB2 on CICS mainframe environment
  - Messaging – Shadow Direct handled traffic between app server and DB2
- Business Process: School Code Search
- Goals: The goal of this test was to execute School Code Search script and determine how Shadow Direct performs under load.

Summary of Results:
1100 users level was applied to the configuration. At 1100 level, both web servers CPUs were fully utilized. It was not possible to generate more load on Shadow Direct because of the bottleneck in the front end. CICS was able to handle 1500 transactions per minute. In order to achieve higher transaction rate, the performance environment needs to retrieve its web server (loaned to production) and remove image hits from the script.

Conclusions-Cycle 23

- The purpose of this test was to determine if FAFSA 7.0 and PIN could run in a shared environment. The test was completed successfully.
- The School Code Search test to stress Shadow Direct is in progress.

4.25 Performance Test with MQ

Test Scenario:
MQ was configured to have the conversion occur on the midrange servers. Initial testing showed that this moved CPU utilization from the mainframe to the midrange - but the impact to the midrange was minimal and deemed acceptable.

A stub was used to generate W412 "edit/submit" transactions. This is the most CPU intensive transaction and was therefore used to create the most intense back-end scenario.

Test Results:
"The interval ended at 09:23 represents "best throughput" of the MQ variant of iterative W412 transactions. Transaction rate was 77per second at 70% processor utilization. Average CICS CPU time was .01265."

The observed limitation was CPU - but the actual CPU available was limited by the number of production processes running which will always get priority over the Performance Test environment. Extrapolation, assuming all resources available to the environment, show a limit of ~111 transactions/sec.

Recommendation:
The group felt that both Shadow Direct and MQ Series would be able to accommodate the peak load of 100-125 transactions/sec. This is based on the fact that peak will be a mix of transactions some of which will be less intensive than the W412's which is expected to handle ~111 transactions/sec.

The group recommends that MQ remain in production for peak with a clearly documented and tested switchover capability for Shadow Direct in place. This decision is to be reviewed based on additional performance testing for both MQ and Shadow and a final review of production peak estimates. Options to increase the "cushion" for peak on MQ will also be evaluated during the additional performance testing.

5 Shadow Direct Performance Tests

FAFSA 7.0 performance test was conducted from September 2002 to December 2002. FAFSA architecture consists of IHS web servers, WebSphere application servers, MQ Series and CICS as a middleware, DB2 and Oracle database. There was a concern that the MQ and CICS might not be able to handle FAFSA peak volume (last week of February). Thus, additional performance tests were planned with Shadow Direct to test FAFSA's backup architecture in January 2003. If there is a need Shadow Direct will be used as a backup during the FAFSA peak.

The objectives for this series of tests are:

1. Validate that Shadow Direct can handle FAFSA's peak volume (i.e. 125 transactions per second). The peak volume needs to be generated by the stub test.
2. Validate the Shadow Direct configuration using the Load Runner test. Load Runner can generate about 25% of the peak volume because of license and hardware constrains.
3. Verify that the application will scale.
4. Validate the N-Tiered architecture for FAFSA by testing four business processes (Fill out FAFSA 7.0, TempSave, Fill out FAFSA 6.0, and FAFSA Correction).

5.1 Performance Test – Cycle 1

Test 1

- Shadow Direct Performance Test Cycle 1 was executed on January 7, 2003.
  - Database – DB2 on CICS mainframe environment
  - Shadow Direct handled traffic between app server and DB2
- Business Process: W12 transaction
- Goals: The goal of this test was to determine if Shadow Direct could handle 125 transactions per second workload to CICS.

Summary of Results:
Load was generated using the Shadow test stub originated from Pearson. 125 transactions per second were achieved during the test for a short period of time. CPU utilization on the Main Frame was at 97% and for which, 57% was from W12 transactions. Performance test and Production environments share the same CPS Main Frame. Shadow configuration was modified during the test in order to send higher load into CICS. The configuration had 15 Shadow pipes at the starting of the test. As the throughput limit was reached, additional pipes were added. 25 pipes in total were used at the end of the test.

Test 2

- Shadow Direct Performance Test Cycle 2 was executed on January 7, 2003.
- Hardware & configuration:
  - Web server – HPL14 & HPL17 (8-way, 440 MHz, 8 GB memory. Max client set to 2048)
  - App server – HPN8 (8-way, 750 MHz, 8 GB memory. Each of the 2 clones had Web thread count set to 40 and heap size set to 1 GB)
  - Database – DB2 on CICS mainframe environment
  - Shadow Direct handled traffic between app server and DB2, 25 Shadow pipes to CICS
- Business Process: W12 transaction, Fill Out a FAFSA 7.0, and Fill Out a FAFSA 6.0
- Goals: The goal of this test was to determine if Shadow Direct could handle 125 transactions per second workload to CICS.

Summary of Results:
Load was generated from two sources: Test stub and LoadRunner. On average, Shadow was able to push between 100 and 120 CICS transactions per second through out the test. At 106 CICS transactions per second level, 90% Mainframe CPU was utilized for which 60% of it was from both the stub and LoadRunner. Performance test and Production environments share the same CPS Main Frame.

Conclusions-Cycle 1

- Shadow Direct was able to meet the test goal of 125 transactions per second briefly.

5.2 Performance Test - Cycle 2

- Shadow Direct Performance Test Cycle 2 was executed on January 9, 2003.
- Hardware & configuration:
  - Web server – HPL14 & HPL17 (8-way, 440 MHz, 8 GB memory. Max client set to 2048)
  - App server – HPN8 (8-way, 750 MHz, 8 GB memory. Each of the 2 clones had Web thread count set to 40 and heap size set to 1 GB)
  - Database – DB2 on CICS mainframe environment
  - Shadow Direct handled traffic between app server and DB2, 25 Shadow pipes to CICS
- Business Process: W12 transaction, Fill Out a FAFSA 7.0, FAFSA Corrections 7.0, and Fill Out a FAFSA 6.0
- Goals: The goal of this test was to determine if Shadow Direct could handle 125 transactions per second workload to CICS.

Summary of Results:
The test stub was run in the beginning of the test. At 40 test stub threads, CICS had 104 transactions per second with 60% CPU. DB2 threads high water mark was at 21 and its service time was at 9 ms. No additional through put gained from increasing the number of test stub threads. 2000 Load Runner users were then added to the mix. CICS transactions still stayed at a constant rate of 100 transactions per second. A transaction limit of around 100 per second was experienced.

The test was stopped to make test stub change. W12 (edit/submit) transaction was substituted with School Code Search (W30) transaction. W30 is a less DB2 intensive transaction. After the change, CICS was able to do 363 transactions per second with 62% CPU.

Prior to the Shadow test, EAI experienced DB2 table lock in its MQ test using the same W12 transaction. During the Shadow test, DB2 table locking was also seen but not as severe as in the EAI test. CSC and Pearson are looking into the cause of DB2 table lock and the implication of it.

Conclusions-Cycle 2

- DB2 table locking issue arose. CSC and Pearson will investigate the source and implications of this issue.
- Shadow was just able to maintain 104 transactions per second with 60% CPU utilization on the mainframe.

5.3 Performance Test - Cycle 3

- Shadow Direct Performance Test Cycle 2 was executed on January 9, 2003.
  - Database – DB2 on CICS mainframe environment

- Shadow Direct handled traffic between app server and DB2, 25 Shadow pipes to CICS, Shadow connections pool was varied from 12 to 36 throughout the test
- Business Process: W12 transaction
- Goals: The goal of this test was to determine if Shadow Direct could handle 125 transactions per second workload to CICS.

Summary of Results:
Load was generated using the Shadow test stub originated from Pearson. 125 transactions per second were achieved with 12 Shadow connections in the beginning of the test. The number of Shadow connections was increased in an effort to increase the number of transactions per second. The result was degradation in the number of transactions to a range between 80 - 90 transactions per second. CSC applied IBM-supplied tuning recommendations, which did not improve CICS transaction rate. CPU utilization on the Main Frame was at 90% of which 49% was from W12 transactions. 25 pipes in total were used for the purpose of this test. Shadow Direct could not maintain 125 transactions / sec. The average transaction rate per second for this test was 100.

Conclusions-Cycle 3

- CICS unable to process entire volume of transactions generated by Shadow test stub; increasing the number of connections reduces the number of transactions.
- Shadow was able to hit 125 transactions per second but unable to maintain this transaction rate.

5.4 Performance Test - Cycle 4

- Shadow Direct Performance Test Cycle 2 was executed on January 9, 2003.
- Hardware & configuration:
  - Web server – HPL14 & HPL17 (8-way, 440 MHz, 8 GB memory. Max client set to 2048)
  - App server – HPN3 (8-way, 360 MHz, 8 GB memory) &HPN8 (8-way, 750 MHz, 8 GB memory. Each of the 2 clones had Web thread count set to 40 and heap size set to 1 GB)
  - Database – DB2 on CICS mainframe environment
  - Shadow Direct handled traffic between app server and DB2, 25 Shadow pipes to CICS, 80 Shadow connections pool, 128 MB JVM
- Business Process: Fill Out a FAFSA 7.0 [Test 1]. FAFSA TempSave-Restore, Fill Out a FAFSA 7.0, and Fill Out a FAFSA 6.0 [Test 2]
- Goals: The goal was to test MQ / Shadow Direct switchover procedure, and determine if Shadow Direct could support 125 transactions per second of workload to CICS.

Summary of Results:
Load was generated using the Shadow test stub and LoadRunner. The Shadow test stub deployed a diverse set of transaction types. With 12 Shadow connections at the beginning of the test, the CICS transaction rate ranged between 65 -70 per second. No backlog of transactions was reported on CICS. LoadRunner virtual users failed in large numbers, due to mismatched application code. The application code on HPN3 and that on HPN8 was not identical. The test was halted as the application code updates took place. Upon restarting the test, the Shadow stub transaction volume was reduced. Once again, LoadRunner recorded a high number of virtual user failure. This was due to a series of DPL Bridge outages, which is an issue that is being investigated. The Bridge outage contributed to a significant reduction of CICS transaction rate, which was 20 per second at the conclusion of testing. Due to various failures during the test, the testing for switching MQSeries / Shadow Direct was inconclusive. In addition, during the second test it was discovered that the stored procedure for FAFSA 6.0 were deactivated, which led a large number of LoadRunner virtual user failure.

A switch over test was conducted later with 50 users. Network Dispatcher redirected traffic from a group of web servers to another before switching to Shadow. Thus, the switchover test was conducted successfully without losing a single user.

Conclusions-Cycle 4

- Application code on servers was not identical; DPL Bridge outages occurred during testing.
- Retested with 50 users. The switchover test was completed successfully. No users were lost.

5 Capacity Planning

The capacity planning and performance test efforts work hand-in-hand. Actual peak data from last year was used to come up with targets for the number of concurrent users to test. Based on the results of the tests, the capacity and utilization requirements were estimated for the following year. Many of the calculations and estimates are based on an assumption that there will be a 50% increase from FAFSA 6.0 in utilization during peak 2003.

5.1 Number of Users

The following tables specify the number of expected users for FAFSA 7.0. Included are tables for the average daily number of users and hits, as well as the estimated peak number of users and hits, and several significant calculations based on the peak number of users estimated.

Assuming 50% growth from FAFSA 6.0

	FAFSA 7.0 Predict (17 hours)	FAFSA 7.0 Predict 4% Increase
Hits/day	105,000,000	
Hits Homepage/day	10,500,000	
Page Views/day	21,000,000	
Users/day	980,000	
User session length (min)	15	
Temp Apps/day	720,000	
App Submits/day	127,500	

FAFSA 7.0 Peak Hour Assumptions

Hits/hour	5,558,824	5,786,735
Hits Homepage/hour	555,882	578,674
Page Views/hour	1,111,765	1,157,347
Users/hour	51,882	54,010
User session length (min)	15	15

Column 1 -(90% of hits during 17 hour period 9am-2am, distributed equally)

FAFSA 7.0 Peak Hour Calculations

Hits/sec	1,544	1,607
Hits homepage/sec	154	161
Page views/sec	309	321
# concurrent users	12,971	13,502
Hits/user	107	107
Page views/user	21	21

Capacity Planning Estimates Table for FAFSA 6.0 and FAFSA 6.0 Actual (from the web trend reports)

	FAFSA 6.0 Predict (17 hours)	(24 hours)	FAFSA 6.0 Actual (17 hours)
Hits/day	70,000,000		61492052.5
Hits Homepage/day	7,000,000		
Page Views/day	14,000,000		11378108
Users/day	700,000		636204.00
User session length (min)	25		14
Temp Apps/day	700,000		480000
App Submits/day	70,000		85000
Hits/hour	3,705,882	4,417,000	4136891.5
Hits Homepage/hour	370,588	441,700	
Page Views/hour	741,176	883,400	
Users/hour	37,059	44,170	37423.76471
User session length (min)	25	25	14
			Column 3 represents Peak Hour of FAFSA 6.0 Actual 2/28/02 @ 21:00
Hits/sec	1,029	1,227	1149.136528
Hits homepage/sec	103	123	
Page views/sec	206	245	
# concurrent users	15,441	18,404	8733
Hits/user	100	100	
Page views/user	20	20	
user think time (sec)	75	75	

PIN Capacity Planning:

The following table shows the existing PIN database volume per hour (maximum):

Business Processes	Jan	Feb	Mar	Apr	May	Jun	July	Aug
Authentication	N / A	14882	7807	8254	9572	6949	8844	10140
Registration	N / A	2306	1688	1034	1182	973	1311	1477

After getting the projected volume from other applications for their future releases, the ITA team determined that the PIN should be performance tested with the following number of transactions per hour.
Registration: 3,459 transactions per hour (50% growth)
Authentication (FAFSA): 18,975 transactions per hour
Authentication (Other Applications): 3,189 transactions per hour
Authentication (Web Services): 1,000 transactions per hour

5.2 Extrapolations

The spreadsheet in diagram 2.1.1 is the method used to extrapolate the number of application servers needed for peak 2002 processing.

Diagram 2.1.1

Server Extrapolations	
Users Tested	3,000
Clones Used	4
Users per Clone	700
Projected # Peak Concurrent Users	13,500/hour
Clones Needed for Peak	16 for FAFSA 3 for PIN
Projected clones per server (8 x 750)	2
# WAS Servers for peak (20% contingency) (8X750 CPU)	10
# IHS Servers for peak (20% contingency) (8X360 CPU)	10

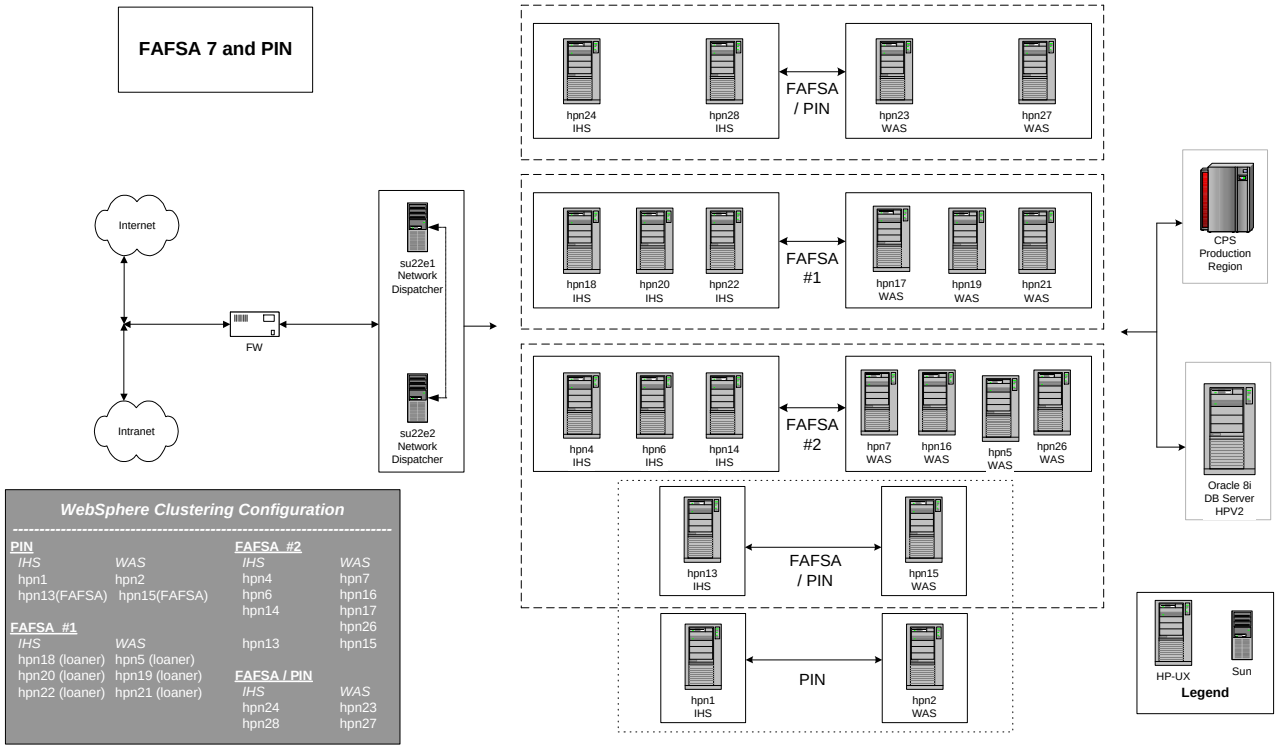
6 Environment Recommendations

Based on the performance testing and tuning, the following configuration should be able to handle the FAFSA peak load with 13,500 concurrent users.

- Web Servers: CPU: 36,000 Mhz, Memory: 40 GB Total
- Application Servers: CPU: 75,000 Mhz, Memory: 80 GB Total

Diagram 2.1.2 illustrates the projected production logical technical architecture requirements.

Diagram 2.1.2



7 Conclusion

7.1 Ongoing Performance Testing

Performance tuning is an on-going process. A system that has been tuned does not guarantee that the tuning enhancements will always be suitable for a growing system. Everyday occurrences such as changes in usage patterns, data volumes and distribution, and applying patches and fixes can make a marked impact on the overall performance of a system. It is important to have a production performance strategy in place. The strategy should include proactive performance monitoring and capacity planning.

Appendix 1

8 FAFSA 7.0 Performance Test Issue List

New and Open Action Items

No	Date Identified	Category	Action Item	Responsible	Status	Severity	Target Date
1	10/30/02	PIN	Post performance test cycle 8: PIN Test Harness (hosted on Su35e11) had very slow response time during the PIN performance test. 11/11/02 - Ran out of TCP connections on SU35E11. Bob will work with Keith to set some parameters correctly on SU35E11. 11/18/02 – CSC needs to monitor the network. 12/02/02 – Test harness running out of memory problem has been corrected. Schedule retest, coordinate with Chad Simmons. 12/04/02 – Performance test cycle 15b tested Web Services. The test showed that TestClientServlet does not close the connections, so after running the test for 20 minutes all the users who are using Web Services failed. This issue is being investigated by ITA and Pearson. 12/09/02 – Connections being dropped (left hanging) – 200 connections. 12/18/02 – This was recreated during the performance test. The test ran for 20 minutes and observed hanging connections. ITA team is working with CSC to investigate the possibility of a firewall networking issue. 12/19/02 - The PIN test harness opens the connections but does not close the connection. Matt Kain informed that there is not a way to close a soap connection. CSC's firewall allows only 25,000 connections in the test environment and the connection time out is set to 2 hours. In production the firewall allows 200,000 connections and the connection time out is set to 1 hour. Thus, during the performance test there were 500 users running web services and in about 30 minutes all 25,000 connections were used up. Thus, we were getting slow response time and failure. The ITA team is working with CSC and Pearson to resolve this issue. 12/23/02 - CSC suggested that there is a problem with Firewall code in the performance test environment. CSC is looking into the option of setting the performance test firewall same as production so that timeout is set to 1 hour and the number of connections will be increased. DLSS is not planning to use the PIN Web Services in January.	Bob Wehrle / Chad Simmons	Open	Medium	12/20/02

			1/9/2002 – The Firewall was updated in production environment but not in performance test. ITA is working with CSC to schedule the firewall update in the performance test environment. Once the firewall is updated then web services test needs to re-run.				
2	11/11/02	WAS	WAS Outstanding issue from FAFSA 6.0 Production: Hung process 11/11/02 - Recreated in the performance test environment and the log info was sent to IBM. 12/19/02 – IBM suggested turning some parameters to recreate this issue. In order to recreate this issue CPU on server needed to be at 100% utilization. This problem was recreated and the logs were sent to IBM. This problem could not be recreated by making CPU 80% busy – in order to recreate this issue CPU had to be 100% busy.	Roshani Bhatt / Bob Wehrle	Open	Low	12/20/02
3	11/22/02	Backend (Tuning)	The performance test that was conducted on 11/22 (to stress the CICS and DB2) showed that the backend needs to be configured for the FAFSA peak. The test was ran with the following configuration/protocol. Configuration: CICS MAXTASK=75 DB2 MAX THREADS=15 DB2 Protocol: Close the MQSeries - CICSPTST bridge Generate a number of messages to the MQSeries queue Open the MQSeries - CICSPTST bridge The CPS mainframe will need to deliver 96 MIPS to the FAFSA 7.0 application in order to sustain a throughput rate of 30 transactions per second (30/48 times 70% times 219 MIPS). This leaves 123 MIPS available for other work (approximately last month's peak hour demand). · The CPS mainframe will need to deliver 153 MIPS to the FAFSA 7.0 application in order to sustain a throughput rate of 48 transactions per second (70% times 219 MIPS). This would leave approximately 66 MIPS available for other work, which, depending on the time of day, might not be enough.	Tom Puddicombe	Open	Medium	12/20/02

			· The CICS configuration needs review: · Fewer DB2 threads were available than were transactions wanting to use DB2. It is not known how increasing the number of DB2 threads will affect transaction throughput. It is expected that increasing the number of threads will increase CPU demand, memory demand, and quite possibly I/O rate · Transactions were unable to start in CICS due to the MAXTASK limit. Increasing the value this parameter will increase the number of concurrent transactions. This will increase CPU demand and will also affect the required number of DB2 threads. · Should the FAFSA 7.0 workload be merged into one of the existing production CICS regions, or should it be placed in its own CICS region. This introduces issues of availability in addition to performance. · The DB2 configuration needs review: · How will increasing the number of CICS (FAFSA 7.0) threads affect DB2. · DB2 buffer pool statistics need review to determine how much adjustment to the number of buffers in each of the various pools is required to handle FAFSA 7.0. · What'll be the performance impact of changing CICS's ACCOUNTREC parameter from "NONE" to "TASK". The parameter change causes DB2 to cut one accounting record per CICS task instead of one accounting record per thread termination. There will be more DB2 accounting records, greater statistical detail, easier quantification of the overheads associated with each of the various CICS tasks, but at the cost imposed by the data collection process. 12/02/02 – Tom suggested to change the DB2 threads and run this test again. 12/05/02 – It was decided that LoadRunner with four N-class machines was generating sufficient load to stress the back-end. CPS configuration required: Data is backing up on the CPS MQ instance – specifically the DPL bridge. "Thread" limitation is perceived to be the issue. This issue arose after the DB2 tuning issue was resolved. Next step: Considered starting another instance of the bridge to provide more throughput. The CPS dead lock and VDC network issues prevented a clean test and clear direction for resolution. This issue will be tested during cycle 15c (12/05/02). 12/06/02 – DB2 tuning was performed during cycle 16b. The thread count was increased to 40 and a high water mark of 32 threads was observed. Further DB2 tuning will be carried out. 12/18/02 – A separate stub was created to generate load on the back-end, so the backend can be				

			tuned properly for peak. Running this program produced issues which CSC is investigating. 12/19/02 - Another test was planned outside of the performance test window to stress and tune the backend. CSC is working with Accenture and Pearson to tune the backend. In this test it was discovered that the CICS CPU reached its max and FAFSA might not be able to handle the peak volume with the current backend configuration. 12/23/02 – CSC has contacted IBM MQ/DPL/DB2 SME to tune the backend. 12/27/02 – CSC, Accenture and Pearson had a meeting regarding tuning the backend. It was agreed that CICS needs to support about 125 transactions per second during FAFSA peak (i.e. PAFSA + IDC + other applications). From the FAFSA PRR meeting, the client expressed an interest to do the performance test using Shadow Direct. Roshani is working on a plan to do the performance test with Shadow in January. Scott Gray opened a severity level 1 ticket with IBM and Gary Adams is communicating with IBM to get the IBM performance team to review performance test data and make recommendations for additional tuning. 1/07/2003 - FAFSA with MQ test was conducted on 1/6/2003. **Test Scenario:** MQ was configured to have the conversion occur on the midrange servers. Initial testing showed that this moved CPU utilization from the mainframe to the midrange - but the impact to the midrange was minimal and deemed acceptable. A MQ stub was used to generate w412 "edit/submit" transactions. This is our most CPU intensive transaction and was therefore used to create the most intense back-end scenario. **Test Results:** "The interval ended at 09:23 represents "best throughput" of the MQ variant of iterative W412 transactions. Transaction rate was 77/second at 70% processor utilization. Average CICS CPU time was .01265." The observed limitation was CPU - but the actual CPU available was limited by the number of production processes running which will always get priority over the Performance Test environment. Extrapolation, assuming all resources available to the environment, show a limit of ~111 transactions/sec. A cycle of Shadow Testing was also performed. "The interval ended 11:08 represents "best throughput" of the Shadow Direct variant of iterative W412 transactions. Transaction rate was 82/second at 58% processor utilization. Average CICS CPU time was .00764." Extrapolation, assuming all resources available to the environment, show a limit of ~136 transactions/sec.				

			Recommendation: The group felt that both Shadow Direct and MQ Series would be able to accommodate the peak load of 100-125 transactions/sec. This is based on the fact that our peak will be a mix of transactions some of which will be less intensive than the w412's for which we expect to be able to handle ~111 transactions/sec. The group recommends that MQ remain in production for peak with a clearly documented/tested switch-over capability for shadow in place. This decision is to be reviewed based on additional performance testing for both MQ and Shadow and a final review of production peak estimates. Options to increase the "cushion" for peak on MQ will also be evaluated during the additional performance testing. And, extrapolation of initial Shadow Testing show a limit of ~136 transactions/sec making it a feasible back-up if MQ thresholds are exceeded. **Next Steps:** - Create ECM's in production to move conversion from the Mainframe to the MidRange - updates planned for Sunday, 1/12. - Continue with scheduled Shadow Direct Testing. - Set up procedure for switching between MQ and Shadow - Would need to be able to execute quickly and be previously tested. - Continue with MQ Testing - to include a test during lowest production loads to further verify peak capacity. - Review additional MIPS necessary to increase peak support cushion - Data available 1/10. - Review production numbers to date and re-evaluate peak estimates to ensure comfort level. 1/09/2003 - Two performance tests were done using Shadow (Load Runner and Stub). The first performance test (1/7/03) got as high as 125 transactions per second at one point. The goal of the test was to reach 125 transactions per second continuously. The second performance test (1/9/03) got as high as 104 transaction per second. Tom, Walton, and Nancy are investigating the reason for not getting higher than 104 transactions per second.				

Completed Actions

No	Date Identified	Category	Action Item	Responsible	Status	Severity	Date Resolved/Due
1	9/17/02	Oracle	Provide Rich Ryan with change request for the session database to be converted to 16k	Roshani Bhatt	Closed	Medium	9/19/02
2	9/17/02	Admin	Send out an updated version of the Performance Test Plan with comments from the meeting.	Roshani Bhatt	Closed	Low	9/18/02
3	9/23/02	DB2	Build, load, and test DB2 for FAFSA performance test environment => 4:00 PM on 9/25/02	Gabriel Perez	Closed	Medium	9/25/02
4	9/25/02	Admin	Set up meeting to discuss performance test plan.	Mike Cline / Roshani Bhatt	Closed	Low	9/25/02
5	9/25/02	Admin	Fill out a FAFSA Renewal functional script creation is problematic, as the code is not stable in the environment.	Matt Kain	Closed	Low	9/26/02
6	9/30/02	Admin	Determine number of records available for Renewal business process => ~1350 records	Bill Schulte	Closed	Low	10/7/02
7	9/30/02	PIN	Determine where invoker web services will be placed	Matt Kain / Jeff Farris	Closed	Low	10/10/02
8	9/23/02	MQ	Problem with MQ/CPS/CICS is NCS code is generating error messages. Scott will work with NCS to implement solution. This was caused by data type mismatch. The root cause of data mismatch was NCS was testing some functionality and was sending bad data to the CICS bridge. This code is fixed and we have not seen the error.	Scott Gray / Dan Butler	Closed	Low	10/7/02
9	9/25/02	MQ	Scott Gray to write formal email documenting the MQ/CPS/CICS problem	Scott Gray	Closed	Low	10/7/02
10	9/30/02	Admin	Update Performance Test Plan high-level document following discussion about schedule	Roshani Bhatt / Mike Cline	Closed	Low	10/10/02
11	9/30/02	App Code	School Code Search needs to invalidate sessions: FAFSA Performance test Cycle 1. ITA provided the JSP directive to NCS so that it will not create sessions. NCS used this directive on the school code search Performance test – cycle 3 proved that school code search is not creating sessions.	Matt Kain	Closed	Low	10/10/02
12	10/3/02	WAS	WebSphere recycled during FAFSA performance test cycle 2. Performance test cycle 5 showed that WebSphere did not recycle. AMI was removed and code was added to talk to MQI interface. This fixed the WAS recycle issue	Bob Wehrle / Scott Gray	Closed	Medium	10/18/02

			and code was added to talk to MQI interface. This fixed the WAS recycle issue.				
13	10/14/02	OS	Verify kernel parameter settings from HP – Bill Gardner verified kernel parameters with HP.	Bill Gardner	Closed	Medium	10/21/02
14	10/16/02	MQ	The CICS region was being crashed due to developers putting messages on the queue after the messages had been run through a translator, which is the wrong procedure when using the Java MQI interface. Scott needs CSC's assistance in diagnosing CICS region to determine whether this can happen in other scenarios. CSC to identify someone to work with Scott.	Scott Gray / CSC resource	Closed	Medium	10/18/02
15	10/16/02	Admin	PIN performance test environment needs Verisign certificate installed.	Bill Gardner	Closed	Low	10/29/02
16	10/24/02	Admin	Due to CPS upgrade ITA may not have the ability to use the performance test environment for three and one half business days. This may affect the writing of functional scripts and recording of LoadRunner scripts, which could in turn adversely affect FAFSA performance test schedule. Action Item: Communicate this impact to stakeholders. Resolution: Upgrades completed ahead of schedule.	Roshani Bhatt	Closed	Low	10/29/02
17	10/24/02	PIN	PIN requires a dedicated session database. ITA submitted a change request to CSC. Resolution: Reuse SESSSTG – 8K database.	Rich Ryan	Closed	Low	10/29/02
18	9/30/02	PIN	Provide test harness for PIN performance test. Test harness is ready but PIN perf. test environment issues prevent implementation. Possible solution: 1) Add a servlet to the Sun perf env that LoadRunner (LR) scripts can call with parameter auth. 2) Set up a LR script to call the HTML version of authentication. The LR script will start by pointing the performance test URL. Replace the "https://dev.pin.ed.gov:8443" with what ever the PIN Perf env URL is going to be. The successful and unsuccessful params can be left alone. Hence, the user will start right on the PIN auth HTML page. Enter your stable data and pin and then you will just be redirect back to the FAFSA home page. 10/30/02- PIN test harness was tested.	Matt Kain / Bob Wehrle	Closed	Low	10/30/02

19	10/11/02	WAS	WebSphere dropped the session table during FAFSA peak: This is an outstanding WAS issue from production. IBM provided the fix for WAS 3.5.6 and it did not work - Perf. Test cycle 4. Bob will work with IBM to get the right fix. 10/29/02 – This was tested with a new patch provided by IBM. During test cycle 8. We were able to recreate this issue without the fix. Applied the fix and ran the test. Session table did not drop. This issue is closed. However, it will be monitored throughout the FAFSA performance test.	Bob Wehrle	Closed	Medium	10/30/02
20	10/28/02	Admin	eSignature business process was not tested as part of the performance test effort for FAFSA 6.0. No issues have been reported in production. The benefit of performance testing eSignature business process for FAFSA 7.0 may be marginal. ITA proposes to substitute eSignature performance with Fill Out a FAFSA 6.0 business process. Got approval from ED.	Roshani Bhatt	Closed	Low	10/31/02
21	11/01/02	Admin	Roshani presented idea of replacing eSignature with Fill Out a FAFSA 6.0 option to ED. ED approved this request.	Roshani Bhatt	Closed	Low	11/15/02
22	11/06/02	WAS	Post-Cycle 9: Configuration: One web server and one application server (both N class machines)- JVM 1GB. We were running four scripts with the following distribution: Fill out FAFSA 7.0: 15%, FAFSA Correction 7%, FAFSA - tempSave 70%, Fill out FAFSA 6.0 8% and 1000 users. Java out of memory was observed which indicates a memory leak 11/19: We received this error because there were too many request coming from the webserver. We increased the number of clone from one to two which resolved this issue.	Bob Wehrle	Closed	Medium	11/11/02

23	9/17/02	Infrastructure	Organize a meeting to discuss OC3 link test. Objective of OC3 link test is to ensure peek FAFSA load can be sustained, the following are the test goals: ▪ Hit FAFSA home page with 90K file ▪ Hit FAFSA home page with 6K GIF file ▪ FTP test ▪ Tune web servers (max client, NDD) Chad needs to confirm the test date. Test is currently scheduled for 11/17. 11/11/02: Question raised: ▪ Can this test hit the performance test environment from the Internet? ? The FASFA performance test team has not received any plans regarding this test from CSC. ? Roshani will follow up with Chad. 11/16/02 – Test was conducted by Mercury Interactive in conjunction with CSC. 11/18/02 – Performance test was conducted hitting FAFSA home page with 1000 users yielded 120 Mb/sec throughput, and 147 Mb/sec utilizing a test GIF file with 400 users.	Roshani Bhatt / Chad Simmons	Closed	Low	11/18/02
24	10/14/02	Admin	Report performance test results and compare production requirements. 11/15/02 – Presentation capturing the performance test results was sent to CSC and Pearson. 11/19/02 – The performance test result (preliminary input for peak hardware) was presented to CSC and ED.	Roshani Bhatt/ Mike Healy	Closed	Low	11/15/02
25	9/17/02	Admin	Send Bob Wehrle and Will Brownlow information on WebTrends configuration. Save IHS logs. 10/29/02 – Saved IHS logs and sent to Bill. 11/18/02 – Bill sent logs to Mike Gonzalez and Don Cherry who are looking into this matter. 12/02/02 – Bill stated that WebTrends is capable of handling logs.	Bill Gardner	Closed	Low	12/02/02

26	10/16/02	MQ	Security measures must be implemented to control the number of individuals who may put messages on a queue. Individual access should be controlled with User IDs who may put messages from specific IP addresses. This action is necessary to prevent the potential of one message shutting down the application. This issue is related to issue number 3. 11/18/02 - MQ configuration for IP security exits will be moving through the environments at the same time as the MQI Pooling code. The code is slated for deployment to the performance test environment on 11/22. Pearson needs to provide workstation IP address. 11/21/02 – Code was merged into the performance environment. Ready for performance test. 12/02/02 – IP filtering is in the performance test environment. Code was tested successfully by EAI & Pearson.	Scott Gray	Closed	Low	12/02/02

27	10/24/02	WAS	Garbage collection was taking 3.7 to 7.0 seconds in the following configurations: ▪ Two web servers & one or two application servers ▪ Session database connections = 100 ▪ JVM at 1 GB Investigate reasons for seven-second garbage collection process, post performance test cycle 6 10/30/02 – Test cycle 8 – JVM set at 512 MB, garbage collection at 4 seconds. After garbage collection, session database connections jumped significantly. Need to investigate this with more tests. 11/5/02 - Test Cycle 9 configuration – two-web and two-application servers. One clone on each app server, heap size is 512 Mb. The thread count was reduced from 200 to 25. In production, the web thread count is 25. We were running four scripts with the following distribution:: Fill out FAFSA 7.0: 15%, FAFSA Correction 7%, FAFSA - tempSave 70%, Fill out FAFSA 6.0 8%. The test started with 400 users and we ramped up to 1025 users. During this test we noticed that the garbage collection took about 3.9 seconds and the session database connections did not jump. Session database connections were 49. Thus, reducing the thread count resolve this issue. This issue will be monitored throughout the performance test. 11/18/02 – HPN3 with 8 x 360 MHz was used as an app server in previous tests. Due to this configuration garbage collection time was higher. 12/02/02 – If the heap is 1GB garbage collection takes 7 seconds. Heap size was reduced to 512 MB and the garbage collection took about 4 seconds.	Bob Wehrle / Roshani Bhatt	Closed	Medium	12./02/02

28	11/01/02	Admin	In the performance test environment there are two 'L' class web servers and two 'N' class application servers. The 'L' class machines are causing a bottleneck to the application servers. 11/8/02 – Research & propose an option to obtain the N class machines for the performance test environment. 11/11/02 – Roshani sent the plan to everyone in the FAFSA production implementation team. Had a meeting with the FAFSA production implementation team to discuss the plan to get the 'N' class machines in the performance test environment. 11/14/02 – Jeff Farris sent updated server roll-out plan to FAFSA production implementation group which has step by step instructions to get N-class machines in the performance test environment. ED approved this plan. HPN13 and HPN14 will be re-networked in the performance test environment on 11/24/02. 11/25/02 – HPN13 & HPN15 are in the performance test environment. These machines will be returned to production environment by 12/11/02.	Roshani Bhatt	Closed	Medium	12/02/02
29	10/16/02	MQ	When max channels limit was reached on MQSeries, an EAI Null Pointer error was thrown. 10/29/02 – EAI is developing connection pooling code. This code will handle all the exceptions. This code is scheduled to move in the perf. Test environment by 11/22. 11/21/02 – Code merged on 11/21. This handles the null pointer exception. Ready for performance test. 12/03/02 – Performance test cycle 15a showed that this issue is not completely fixed. The following message was observed "array out of bound error." EAI team is working on this code. Scheduled to be in the performance test environment by 12/06/02. 12/06/02 – This was tested in cycle 16b successfully.	Scott Gray	Closed	Medium	12/09/02

30	11/07/02	Oracle	Post-Cycle 10: Student Access creates sessions which are 18K. FAFSA session database is 16K. Row chaining issues were observed. CPU on the application server was running high (above 90% utilized). Research and present options (e.g. reducing the session size). 11/11/02- Actual session size is 8.8k (when serialize to a file) however when the session reach to the Oracle DB (sessions are padded and the session size becomes 18k). Session database is 16K thus when the session size becomes more than 16K we see row chaining. ITA will work with IBM support to find out resolution to this issue. Pearson will do some research to reduce session size for Student Access. Student Access script performs PDF generation. This was causing high CPU utilization on the application server. Pearson suggested that the LoadRunner script should be changed so that it will generate HTML file instead of PDF file. The script will be changed and we will test this by 11/25. 11/18/02 – Allocate more database space, Rich Ryan indicated that an additional database exists for FAFSA 7.0 that can be used for this purpose. 11/25/02 – Matt and Bob are working on what is being put on Oracle and why sessions are being padded. 12/09/02 – Allocating more space for session database for FAFSA 7.0 for next year (FAFSA 8.0), the application should reduce the session size.	Bob Wehrle / Matt Kain	Closed	Medium	12/09/02
31	12/06/02	MQ	Test cycle 16b – Messages were expiring on the bridge. This resulted in LoadRunner user failures. After some investigation it was found that FAFSA application has a 15 second time frame to pick up the message and bridge sees the time value as 1.5 seconds. As a result of this condition, messages were not queuing but expiring. The time conversion piece of the EAI code needs to be changed, so that the message can stay alive for 15 seconds rather than 1.5 seconds.	Scott Gray	Closed	Medium	12/09/02
32	9/25/02	Infrastructure	Admin server is killed every 30 minutes performance environment: post-test Cycle 1 – in progress 11/18/02 – Dana Webb is researching the following issue: Keeping connection alive through firewall, increasing the TCP parameter. 12/09/02 – A parameter change fixed the killed server issue. 12/16/02 – ITA team has been dumping the file every 15 minutes to keep the admin server alive and will continue to do this for the rest of the performance test cycles.	Bill Gardner	Closed	Low	12/16/02

33	11/26	MQ	12/02/02 – Deadlocks were seen during the performance test. The issue is a dead lock on CPS MQ processes – MQ logging dead lock causes stop to all messages. This issue is being handled by CSC, hence a GCARS should be logged. The patch is scheduled for installation on 12/08/02. 12/09/02 - Patches regarding the deadlocks were applied on 12/08. CSC and EAI expect this patch to solve this issue. 12/15/02 – Ran 5 performance tests after the patches were applied and the deadlocks were not seen.	Nancy Matesia/Scott Gray	Closed	Medium	12/16/02
34	12/10	WAS	Observed Java out of memory error on 12/12/02. There were 3000 users running in the system. WebSphere and IHS were not tuned. 12/12/02 – Tuned WebSphere. Increased JVM from 512 MB to 1 GB. 12/13/02 – Ran a test with 3000 users with two 1GB JVMs. No out of memory errors were observed. This issue will be monitored throughout the performance test.	Bob Wehrle	Closed	Medium	12/16/02

35	9/23/02	MQ	Find cause for MQ/CPS/CICS intermittent problem. Problem identified by Scott Gray (EAI). Scott is working with IBM to obtain the eFix. 10/29/02 - GTF traces will be formatted by CSC and sent to IBM. A conference call with IBM is scheduled for 11/5 11/11/02 - IBM needs the GTF traces reformatted. Scott will work with CSC to get this trace ready and sent to IBM. 11/18/02 – on 11/15 GTF traces were redone and sent to IBM for review. 11/25/02 – Scott received patch from IBM and will work with CSC to schedule the install. 12/02/02 – Patch is scheduled to be in the performance test environment on 12/08/02 – ECM 853. 12/08/02 – This patch was applied and backed out due to an issue. Logs were collected to send to IBM. 12/16/02 – A new patch was received from IBM. This patch was applied in the performance test environment on 12/16. 12/17/02 – Once the patch was applied Pearson and EAI tested this scenario by doing the following: ▪ Sent right data – data passed through ▪ Sent the reverse byte bad data – an error was received ▪ Sent the right data – data passed through It did not require the recycle the CICS region.	Scott Gray	Closed	Medium	12/17/02

36	11/11/02	App Code	Perf test cycle 11: Check status business process creates sessions. JSP creates session by default so a large number of sessions were created in the session database. Need to add JSP directive on the check status business process. Cycle 13 (11/19/02) – This business process was tested again and it was found that check status was not fully invalidating the sessions. The check status business process creates sessions but when a user clicks the exit button, it nulls the session rather than invalidating it. Thus the session remains in the database with size 0. Once the code is fixed this business process will be tested again. 12/17/02 – Tested this business process with 500 users. There were only 100 sessions in the database, hence the sessions were invalidated correctly.	Matt Kain	Closed	Medium	12/17/02
37	12/10/02	WAS	JSP – more than one clone on a server. Take a failure on 'init' trying to initialize JSP. Need to add a script to compile the JSP, WAS 3.5.6 comes with such a script. 12/17/02 – When using more than 2 clones are utilized – one clone will receive an error if the other one is trying to compile. To pre-compile a JSP file in a Web application in WebSphere there's config file in the bin directory that must be filled out with a set of parameters. Once the code merge is complete, and the config file is set up, typing the command "JSPBatchCompiler.sh" will re-compile the JSP. This script is to be run whenever WAS is restarted.	Bob Wehrle	Closed	Low	12/17/02

38	12/06/02	Load Runner	Post cycle 16b: The following message was observed in the log file: Java. Lang.NullPointerException FOTWServlet – FOTWServlet.do Post: client's session values : SKIPINDEX : eoe [This value should have been blank] Investigation is required. 12/13/02 – Code was meged to resolve this issue. This issue was observed in the performance Cycle 18b (12/13). More investigation is needed. 12/17/02 – After some investigation it was found that this error was induced by LoadRunner. If there is a problem in the backend, the LoadRunner users jump from one clone to another and corrupt the session. This happened because Fill Out a FAFSA time for the LoadRunner script was set to 7.5 minutes (i.e. each transaction takes couple of seconds to finish). The clone might not have updated information which causes the session to corrupt.	Matt Kain/ Bob Wehrle	Close d	Medium	12/17/02
39	12/06/02	App Code	The system time changes and is padded by two extra zeros. This causes an large number of error messages in the log file: ▪ AFOTW Database pad String (), the length of the value to be padded ('10') is greater than the pad size ('8') ▪ getTime (); submit to mainframe/Oracle check the time-stamp length, "00" throws an error. 12/13/02 – Identified as a threading issue with a date / time stamp. Code was merged to resolve this issue. This issue will be kept open and the performance test team will monitor the log files to ensure its resolution. 12/17/02 – This issue was not seen in the performance test environment on December 16th and December 17th. The FAFSA code merged resolved this issue.	Matt Kain / Bob Wehrle	Close d	Medium	12/17/02

40	12/10/02	MQ	Message expiring on the bridge. 12/12/02 - MQ sync q – causes duplicates. CSC & EAI ran the utility program to resolve this issue. A ticket was opened with IBM for this issue on 12/13/02. During the performance test this issue was not observed. This issue will be monitored in future performance tests. 12/17/02 – This issue was monitored in several performance test cycles. We did not see this issue under high volume. The utility program needs to be run in the production environment so that this will not happen in production.	Scott Gray	Closed	Medium	12/17/02
41	11/11/02	WAS	WAS Outstanding issue from FAFSA 6.0 Production: Slow initialization of application server at start up. 11/18/02 – During FAFSA peak production there were 50 clones. Thus, restarting the app server took a long time. This year – for FAFSA 7.0 we are planning to have at most 15 clones for FAFSA during peak, so this issue would not occur. We will test this configuration with 8 clones in the performance environment. Roshani followed up with Joe Hala to see if this is still happening in the FAFSA 6.0 production environment. 12/16/02: This issue was tested in the performance test environment test environment with 10 clones and the application servers started within reasonable time (within 2 minutes). There were not any timeout messages. Joe Hala confirmed verbally that this is not happening in the production now. Pending email confirmation from Joe. 12/18/02 – Roshani received the confirmation email from Joe Hala indicating that this issue has not been observed in the FAFSA production environment for six months when CSC restarted WAS.	Roshani Bhatt / Bob Wehrle	Closed	Medium	12/18/02

42	9/25/02	Backend	Confirm the performance test dates to generate load on CPS: NCS to determine if mainframe load can be tested in Cycles 8 & 9 (11/12) 11/8/02 – Test Cycle 9 & 10 test: FAFSA correction was ran while the mainframe was running the batch processes to stress the mainframe. Batch processes that were generating load were causing issues, as DB2 locks were observed. Once NCS fixes the Batch process, the mainframe load test needs to be performed. The goal of this test is to measure how CPS would perform under load. 10/30/2002- Post performance test cycle 8: When FAFSA Corrections script was run some users failed during the test. DB2 deadlocks were observed during the test. Time out error was seen in the log file. There was a problem with batch process that Pearson was running. Pearson will fix the batch process and we will run this test again. 11/12/02 – Investigation found a batch processor problem where a cursor is not behaving correctly. Transactions were timing out and the batch processes were failing. An internal High priority ticket was opened. ETA for this ticket to be resolved is 11/13 and we will test this on 11/14. 11/21/02 – Gabe reported that this business process is ready to be tested. 12/02/02 – Correction records were not processed, hence this test could not be run this week. This test is scheduled to run on 12/18. 12/18/02 – Pearson recreated the load in CPS during FAFSA performance test successfully. During this test transaction abends were observed on compute, which is tracked as a separate issue. 12/19/02 – Ran this test again. DB2 service time was 0.02 seconds with 8% CPU utilization; CICS performed 533 transactions per minute. The batch job accessed FAFSA Correction records before the accessing the FAFSA application, thus abends occurred. This is a data loading issue and will not occur in production.	Gabe Perez	Closed	Medium	12/19/02

43	11/26/02	MQ	During Cycle 14b an issue was encountered with an MQ Series abend that IBM believes might be related to APAR PQ58704/UQ64230. There is a PTF available, which Nancy plans to get applied early next week. The subsequent restart of CPA1 was delayed because human eyeballs missed an outstanding message on a console somewhere. MQ came back to life within seconds of that message getting its appropriate reply. Since the answer should be a constant (always "Y"), I'll initiate a request to the automation group to have OPS/MVS respond "automagically". 12/12/02 – Patches were applied, however, during the December 13th test (Cycle 18b) transaction abends were observed. CSC and EAI are working with IBM to resolve this. 12/16/02 - Several tests were run on 12/14 and the information was captured on a trace file and sent it to IBM. IBM provided the fix for abends. 12/17/02 – A new fix was received from IBM, description as follows: “What happens is that CKBR browses a message on the request queue and issues a START for a CKBP, passing the MsgId of the associated message. A Bridge Start Element, BSE, is created and added to the 'starting' chain. CKBR then browses to the end of the queue, and starts the browse again. The CKBP transaction then starts an issues a destructive MQGET (within syncpoint) for the messages based on the passed MsgId. However, CKBR has just had this message returned again, but is waiting to be dispatched on the QR TCB. The CKBP then runs to completion, syncpoints, and then removes the BSE from the 'starting' chain. CKBR is then dispatched on QR and searches for a BSE for the MsgId on the 'starting chain'. As such a BSE is not found, another CKBP is started, which then gets 2033 from the MQGET which results in the CKB5. A fix for this problem has been coded (changed CKBP so that it leaves the BSE on the 'starting' chain; CKBR will then remove the BSE when it searches the 'starting' chain'). As mentioned previously, I have recreated this problem on our test systems and have confirmed that the CKB5 abends to not occur with the fix applied. I have sent you a usermod containing the fix in anticipation that the customer will apply and provide feedback.” This fix was applied in the performance test environment and several performance tests were conducted. The abends were recreated. CSC and EAI are working with IBM to resolve this issue.	Tom Puddicombe	Closed	Medium	12/19/02

44	12/13/02	MQ	Timeout exception: No reply received within specified timeframe MQJE001: Completion Code 2, Reason 2033. This is not an EAI issue – EAI code is just capturing the timeout error. This will be seen any time anything causes a reply not to be received in 15 seconds interval specified by the FAFSA application. During the performance test DB2 was maxing its thread and transactions abends were seen which may have caused the timeout. This issue needs to be investigated. 12/17/02 – After some investigation it was discovered that, for unknown reasons, DB2 service time was increasing significantly which causes the DB2 thread count to be maximized at its limit of 40, which causes an error. Tom is investigating this issue with Pearson mainframe personnel. 12/27/02 –A PTF was received from IBM and applied in the DPL Bridge on the week of December 20th. Several performance tests were run after applying this PTF and the timeout errors were not seen.	Tom Puddicombe	Closed	Medium	12/20/02
45	12/13/02	Admin	Image Data Capture (IDC) stress test – encounter issues generation high load. 12/17/02 – Roshani and Bill worked to fit the IDC stress test in the performance test schedule for December 19th and December 20th. 12/19/02 – IDC ran their stress test during FAFSA performance test and encounter some issues to generate the load. IDC ran their test on Thursday afternoon. The IDC stress test is outside the scope of this performance test so this issue is closed. 12/27/02 – Scott Meyer updated Roshani. IDC's target was 5500 documents per hour during peak. The majority of the processes are able to achieve above 9,000 documents per hour range so this issue is closed.	Bill Schulte	Closed	Low	12/20/02

8.1 FAFSA 7.0 Performance Test Issue Summary

Category	Total Issues	High Priority	Medium Priority	Low Priority
PIN	4	0	1	3
WAS	8	0	6	2
Backend Tuning	2	0	2	0
Oracle	2	0	2	0
Administrative	13	0	1	12
DB2	1	0	1	0
Load Runner	1	0	1	0
Infrastructure	2	0	0	2
MQ	11	0	8	3
Application Code	3	0	2	1
Operation System	1	0	1	0
Totals	48	0	25	23